How to count words?

In this lecture we want to count words and for this we have to ask ourselves what a word actually is? We will learn different methods to compare words and get an insight into the linguistic sub-discipline of lexicography and morphology. We will put this knowledge into a transducer that will enable us to normalize texts and gather statistics about words. Finally, we discuss how our solution is transferable to other languages, such as Chinese.

Frequency of words

Text:

In this lecture we want to count words and for this we have to ask ourselves: what are words actually? We will learn different methods to compare words and get an insight into the linguistic sub-discipline of lexicography and morphology. We will put this knowledge into a transducer that will enable us to normalize texts and gather statistics about words. Finally, we discuss how our solution is transferable to other languages, such as Chinese.

- Should „We“ and „we“ count as the same word?
- Should „is“ and „are“ be considered equal?
- …

Frequency of words

Language:

„I do uh main- mainly business data processing.“

„Seuss’s cat in the hat is different from other cats!“

- „uh“: should we also count speech disfluencies?
- „main-“ How to count fragments?
- What about plural –s?

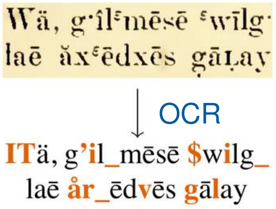
Fuzzy String Matching

Technique of finding strings that match a pattern approximately

https://en.wikipedia.org/wiki/Approximate_string_matching

Why Fuzzy String Matching?

- **Optical Character Recognition (OCR) errors:**



- **Spelling Errors:**
  - upper / lower casing,
  - Typing errors,
  - …
- **[Phonetically ambiguous words](): e.g. “to”, “too”, “two”**
- **Pronunciation complicated or transcription unclear:**
  - “Supercalifragilisticexpialidocious“

    Pronunciation (IPA): /ˌsuːpərˌkælɪˌfrædʒɪˌlɪstɪkˌɛkspiˌælɪˈdoʊʃəs/
  - Proper names: „Maier“, „Meier“, „Mayr“

How does it sound? -> http://ipa-reader.xyz/

NLP-Technique for dealing with OCR errors: https://shrutirij.github.io/ocr-el/

Example: Spelling Errors

`Gierafe`

`Gieraffe`

`Girafe`

`Girafhe`

Which version is „close“ to the correct *german* version (Giraffe)?

Example: Spelling Errors

`Giraffe`

`Gierafe`

Correct spelling with 7 characters

Error?

Example: Spelling Errors

Giraffe

Gierafe

Correct spelling with 7 characters

1 insertion („e“)
1 deletion („f)

2 errors 2/7 = 0.286

Example: Spelling Errors

`Giraffe`	Correct spelling with 7 characters		
`Gierafe`	1 insertion („e“) 1 deletion („f)	2 errors	2/7 = 0.286
`Gieraffe`	1 insertion („e“)	1 error	1/7 = 0.143

Example: Spelling Errors

`Giraffe`	Correct spelling with 7 characters		
`Gierafe`	1 insertion („e“) 1 deletion („f)	2 errors	2/7 = 0.286
`Gieraffe`	1 insertion („e“)	1 error	1/7 = 0.143
`Girafe`	1 too few („f)	1 error	1/7 = 0.143

Example: Spelling Errors

Giraffe	Correct spelling with 7 characters		
Gierafe	1 insertion („e“) 1 deletion („f)	2 errors	2/7 = 0.286
Gieraffe	1 insertion („e“)	1 error	1/7 = 0.143
Girafe	1 too few („f)	1 error	1/7 = 0.143
Girafhe	1 substitution („h“ instead of „f“)	1 error	1/7 = 0.143

Example: Spelling Errors

`Giraffe`	Correct spelling with 7 characters		
`Gierafe`	1 insertion („e“) 1 deletion („f)	2 errors	2/7 = 0.286
`Gieraffe`	1 insertion („e“)	1 error	1/7 = 0.143
`Girafe`	1 too few („f)	1 error	1/7 = 0.143
`Girafhe`	1 substitution („h“ instead of „f“)	1 error	1/7 = 0.143
		„Edit distance“ or „Levenshtein-Distance“	WER

How to search for similar strings?

Let $(\boldsymbol{U}, \boldsymbol{d})$ be a metric space, i.e. $\boldsymbol{U}$ be our „universe of objects" and $\boldsymbol{d}: \boldsymbol{U} \times \boldsymbol{U} \rightarrow \boldsymbol{R}^{+}$ a distance metric satisfying

- $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{y}) = \mathbf{0} \Leftrightarrow \boldsymbol{x} = \boldsymbol{y}$
- $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{d}(\boldsymbol{y}, \boldsymbol{x})$
- $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{z}) \leq \boldsymbol{d}(\boldsymbol{x}, \boldsymbol{y}) + \boldsymbol{d}(\boldsymbol{y}, \boldsymbol{z})$

Paper „Fast Approximate String Matching in a Dictionary": https://ieeexplore.ieee.org/document/712978

How to search for similar strings?

Let $(\boldsymbol{U}, \boldsymbol{d})$ be a metric space, i.e. $\boldsymbol{U}$ be our „universe of objects“ and $\boldsymbol{d}: \boldsymbol{U} \times \boldsymbol{U} \rightarrow \boldsymbol{R}^{+}$ a distance metric satisfying

- $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{y}) = \mathbf{0} \Leftrightarrow \boldsymbol{x} = \boldsymbol{y}$
- $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{d}(\boldsymbol{y}, \boldsymbol{x})$
- $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{z}) \leq \boldsymbol{d}(\boldsymbol{x}, \boldsymbol{y}) + \boldsymbol{d}(\boldsymbol{y}, \boldsymbol{z})$

Idea

Given a new query $q \in \boldsymbol{U}$ and a maximum distance k, retrieve all strings in our vocabulary $\boldsymbol{V} \subset \boldsymbol{U}$ with a distance at most k from q, i.e.

output all $\mathrm{x}^{*} \in \boldsymbol{V}$: $\boldsymbol{d}(\mathrm{x}^{*}, q) \leq \mathrm{k}$

Paper „Fast Approximate String Matching in a Dictionary“: https://ieeexplore.ieee.org/document/712978

Notes on string edit distances

- **There are different edit distances for string sequences**
- **Not all edit distances satisfy the symmetry relation $\boldsymbol{d(x,y) = d(y,x)}$ of a distance metric**

https://en.wikipedia.org/wiki/Edit_distance

Error Types & Word Error Rate

- **Three types of errors:**
  - I := #Insertions („too much")
  - D := #Deletions („too few")
  - S := #Substitutions („confusion")
  - N := #SymbolsOfCorrectString

- **Above metrics on word level => Word Error Rate**

$$WER = \frac{S + D + I}{N}$$

Input

```
X[1..M], Y[1..N]                        // 1-indexed, of length m and n respectively
```

Initialization

```
d[0..M, 0..N] := zeros()          ⇨     // set all elements in 0-indexed array to zero
For all i: d[i,0] := i
For all j: d[0,j] := j
```

Recurrence Relation

```
For j from 1 to N:
    For i from 1 to M:
```

$$
d[i, j] := \min \begin{cases} d[i-1, j] + 1 & \text{// deletion} \\ d[i, j-1] + 1 & \text{// insertion} \\ d[i-1, j-1] + \begin{cases} 2; \text{ if } X[i] \neq Y[j] \\ 0; \text{ if } X[i] = Y[j] \end{cases} & \text{// substitution} \end{cases}
$$

Termination:

```
d[N,M] is the distance
```

See *Chapter 2.5.1: https://web.stanford.edu/~jurafsky/slp3/ed3book_jan72023.pdf*

Fuzzy String Match:

Grapheme Sequence	Phoneme Sequence	
TO	T UW1	works.
TOO	T UW1	
TWO	T UW1	
Robert	R AA1 B ER0 T	Does not work
Rupert	R UW1 P ER0 T	

Robert => Hash: R163
Rupert => Hash: R163

Wie findet man diesen Hash?

https://de.wikipedia.org/wiki/Unscharfe_Suche
https://de.wikipedia.org/wiki/Phonetische_Suche
http://svn.code.sf.net/p/cmusphinx/code/trunk/cmudict/cmudict-0.7b

- **Robert C. Russell and Margaret King Odell**
- **Patented in 1918:**

1. Retain the first letter of the name drop all other occurrences of a, e, i, o, u, y, h, w.
2. Replace consonants with digits as follows (after the first letter):
   1. b, f, p, v → 1
   2. c, g, j, k, q, s, x, z → 2
   3. d, t → 3
   4. l → 4
   5. m, n → 5
   6. r → 6
3. If two or more letters with the same number are adjacent in the original name (before step 1), only retain the first letter; also two letters with the same number separated by 'h' or 'w' are coded as a single number, whereas such letters separated by a vowel are coded twice. This rule also applies to the first letter.
4. If you have too few letters in your word that you can't assign three numbers, append with zeros until there are three numbers. If you have four or more numbers, retain only the first three.

https://de.wikipedia.org/wiki/Soundex
https://de.wikipedia.org/wiki/Metaphone
https://de.wikipedia.org/wiki/K%C3%B6lner_Phonetik

Text pre-processing

1. **Text translated in tokens: Word segmentation**
2. **Normalisation: gather comparability**
   - Normalizing
   - Upper- and lower-casing
   - Morphology
   - Lemmatization/stemming
3. **Sentence Segmentation**

Tokens vs. Types

Distinguish two ways of talking about words

Token vs. Typen?

1 Kranich/Type

10 Kraniche/Tokens

1 individuum or
„identity“

Charles S. Peirce 1906, analytischen Sprachphilosophie

Token vs. Typen?

Beispiel: „HELLO“

#Tokens: 5

#Types: 4 (here: E, O, H, L,)

1 Kranich/Type

10 Kraniche/Tokens

1 individuum or „identity“

Charles S. Peirce 1906, analytischen Sprachphilosophie

Token vs. Typen?

Beispiel: „HELLO“

#Tokens: 5

#Types: 4 (hier: E, O, H, L,)

Beispiel: „There are cars.“

#Tokens: 3

#Types: 3 (there, are, cars)

?

cars = car?
are = were = be = is?

Charles S. Peirce 1906, analytischen Sprachphilosophie

Typen und Token

Type: an element of the vocabulary

Token: an instance of that type in running text

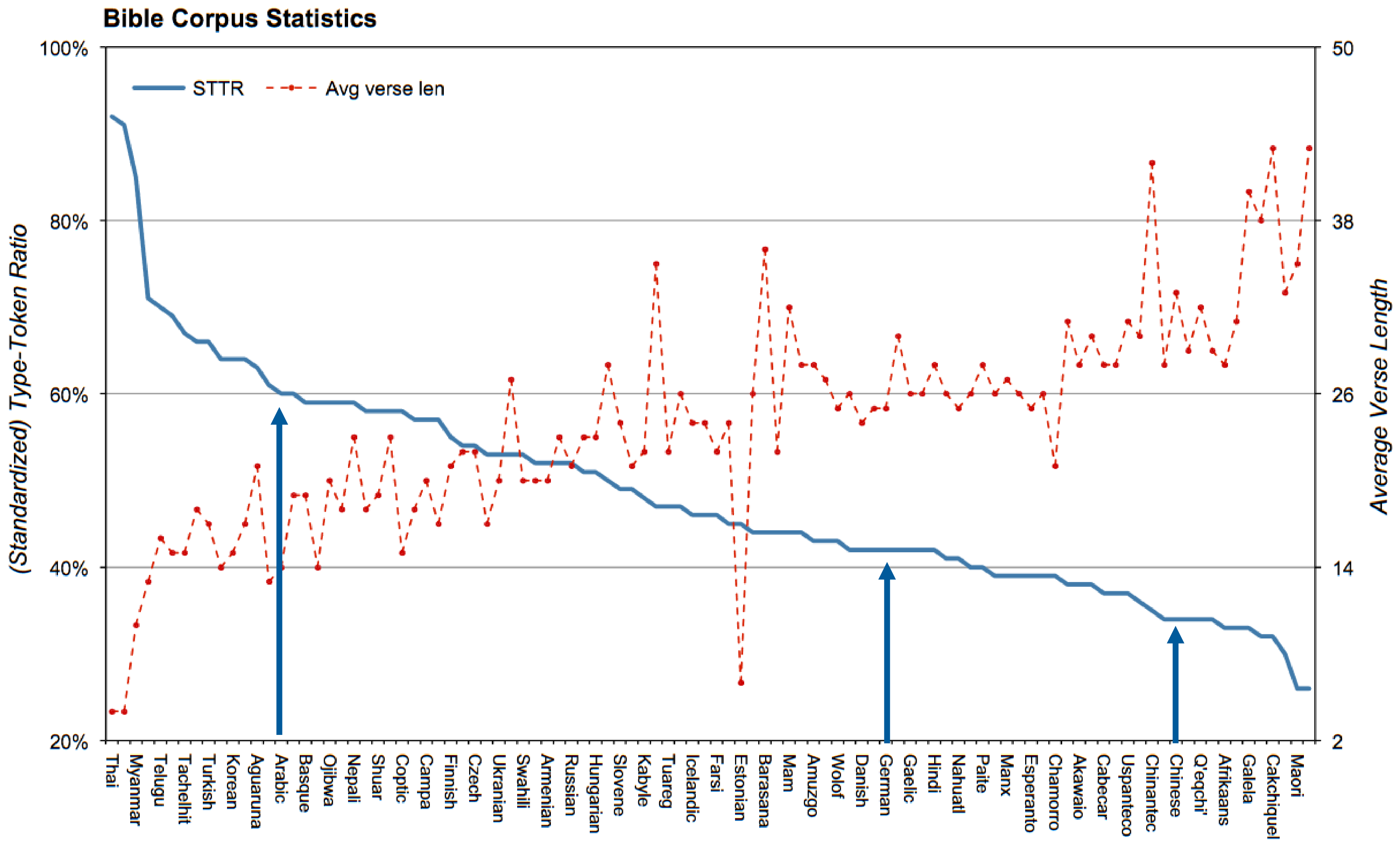
- **Church & Gale (1990): $|\text{Typen}| > O(|\text{Tokens}|^{0.5})$**

	|Tokens|	|Typen| := Vokabular Größe
Switchboard phone conversations	2 400 000	20 000
Shakespeare	884 000	31 000
Google n-gram	1 Trillionen	13 000 000

Further Reading: Chapter 2.2, Jurafsky

https://de.wikipedia.org/wiki/Token_und_Type
https://plato.stanford.edu/entries/types-tokens/#WhaDis

Typen-Token-Ration in verschiedenen Sprachen



Tokenization

Defining words

Segmentation of a text into units on a word level,

aka „words“

- **For German, English etc: ususally simply words separated by whitespaces**
- **But there are special cases**

Special Cases

„Finland‘s capital“	Finland, Finlands, Finland‘s?
What‘re	What are
I‘m	i am
isn‘t	is not
Hewlett-Packard	HP, Hewlett Packard
State-of-the-art	state of the art
Lowercase	lower-case, lowercase, lower case
San Francisco	one token or two?
m.p.h., PhD	?

Special Cases in different Languages

L'ensemble	**L, L', Le?**
L'ensemble	**un ensemble**

Lebensversicherungsgesellschaftsangesteller

⇒ Compound splitter required:

- Leben s
- versicherung s
- gesellschaft s
- angesteller

Slang in Japanese:

フォーチュン500社は情報不足のため時間あた$500K(約6,000万円)

Katakana Hànzì Hiragana Kanji Romaji

https://de.wikipedia.org/wiki/Japanische_Schrift

Slang in Japanese:

フォーチュン500社は情報不足のため時間あた$500K(約6,000万円)

Katakana *Hànzì* Hiragana Kanji Romaji

那是一句话。	(Chinese)
それは一文です。	(Japanese)
นั่นคือประโยค	(Thai)
그것은 문장입니다.	(Korean)
This is a sentence.	(English)

Segmentation into words?

„Most common“: Max-Match Segmentation

Research: Neural nets for word segmentation

https://de.wikipedia.org/wiki/Japanische_Schrift

Max-Match Segmentation

Languages without „obvious“ word boundaries in grapheme sequences

Example

莎拉波娃现在居住在美国东南部的佛罗里达

English: „Sharapova now lives in Florida in the southeast of the United States “

Example

莎拉波娃现在居住在美国东南部的佛罗里达

Longest word in vocabulary? – no.

Vokabulary:
现在
的
国东
在美
莎拉波娃
居住
南部
佛罗里达
佛
罗
里
达
居
住
南
部
...

Example

莎拉波娃现在居住在美国东南部的佛罗里达

Longest word in vocabulary? – no.

```
Vokabular:
     现在
      的
     国东
     在美
   莎拉波娃
     居住
     南部
   佛罗里达
      佛
      罗
      里
      达
      居
      住
      南
      部
      …
```

Example

莎拉波娃现在居住在美国东南部的佛罗里达

$\underbrace{\text{莎拉波}}$

Longest word in vocabulary? – no.

Vokabulary:
现在
的
国东
在美
莎拉波娃
居住
南部
佛罗里达
佛
罗
里
达
居
住
南
部
...

Example

$\underbrace{\text{莎拉波娃}}$现在居住在美国东南部的佛罗里达

Longest word in vocabulary? – yes.

莎拉波娃

Vokabulary:
现在
的
国东
在美
莎拉波娃
居住
南部
佛罗里达
佛
罗
里
达
居
住
南
部
...

Example

莎拉波娃现在居住在美国东南部的佛罗里达

Longest word in vocabulary? – no.

莎拉波娃

Vokabulary:
现在
的
国东
在美
莎拉波娃
居住
南部
佛罗里达
佛
罗
里
达
居
住
南
部
...

Example

莎拉波娃现在居住在美国东南部的佛罗里达

$\underbrace{\text{现在}}$

Longest word in vocabulary? – yes.

莎拉波娃 现在

Vokabulary:
现在
的
国东
在美
莎拉波娃
居住
南部
佛罗里达
佛
罗
里
达
居
住
南
部
…

Example

莎拉波娃现在居住在美国东南部的佛罗里达

Longest word in vocabulary? – no.

莎拉波娃 现在

Vokabulary:
现在
的
国东
在美
莎拉波娃
居住
南部
佛罗里达
佛
罗
里
达
居
住
南
部
...

Example

莎拉波娃现在$\underbrace{\text{居住}}$在美国东南部的佛罗里达

Longest word in vocabulary? – yes.

莎拉波娃 现在 **居住**

Vokabulary:
现在
的
国东
在美
莎拉波娃
居住
南部
佛罗里达
佛
罗
里
达
居
住
南
部
...

Example

莎拉波娃现在居住在美国东南部的佛罗里达

Longest word in vocabulary? – no.

莎拉波娃 现在 居住 ………………

Vokabulary:
现在
的
国东
在美
莎拉波娃
居住
南部
佛罗里达
佛
罗
里
达
居
住
南
部
…

Example

莎拉波娃现在居住在美国东南部的佛罗里达

莎拉波娃 现在 **居住** **在美** **国**东 **南部** **的** **佛**罗里达

Pinyin: Shā lā bō wá xiànzài jūzhù zài měiguó dōngnán bù de fóluólǐdá

…and for English?

```
Thecatinthehat     =>   the cat in the hat
Thetabledownthere =>   theta bled own there
                        (correct: the table down there)
```

Funktioniert nicht für Englisch, Deutsch, …
Wir häufig mit Grammatiken gelöst.

Segmentierung ist aktives Forschungsfeld in allen Sprachen!

Normalization

Remove noise and other superfluous information, establish comparability.

Special Cases

U.S.A. vs. USA

GM vs. General Motors vs. general motors

Fed vs. fed

US vs. us <= context

Define equivalence classes of terms

Examples: Internet Slang

Input	Output
2moro	tomorrow
2mrrw	tomorrow
2morrow	tomorrow
2mrw	tomorrow
tomrw	tomorrow
b4	before
otw	On the way

Examples: Noise

Input	Output	word stem
..trouble..	trouble	troubl
trouble<	trouble	troubl
trouble!	trouble	troubl
<a>trouble	trouble	troubl
1.trouble	trouble	troubl

We'll get to that in a minute!

https://github.com/kavgan/nlp-in-practice/blob/master/text-pre-processing/Text%20Preprocessing%20Examples.ipynb

Summary

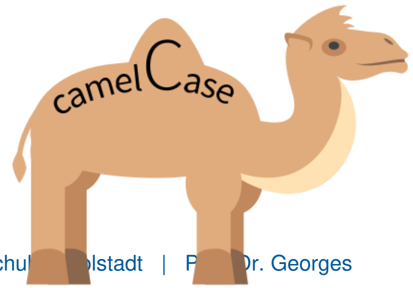
- [!”#$%&’()*+,-./:;<=>?@[\]^_`{|}~]
- Space, line break
- <tr>, <a>, <p>, …

Capitalization

iS uPPeR AnD LoWEr CAsiNg ReaLLy IMportant FoR uNDerStandAbilITY?

TODO

Sentence start/Sentence case	General syntactic agreements
Munich, Audi, United States	Proper names
BMW, ICE, US	Abbreviations
easyJet A319, WikiWord, WikiCase, PhD, BSc., StGB, GmbH, TzBfG, macOSm iPhone, BahnCard, RegionalExpress, InterCityExpress	„Marketing“
I (in English)	Peculiarities of the language
…	



Good to know:
https://en.wikipedia.org/wiki/Title_case

Morphology

The study of the way words are built up from smaller meaning-bearing units

Morphemes

- A ***morpheme*** **is the smallest meaning-bearing unit of a language**
- A ***stem*** **is the central morpheme of the word, supplying the main meaning**
- **Affixes: Bits and pieces that adhere the stems (often with grammatical functions)**

Word Formation

- **Words arise**
- **A new word „unhappy“ can be derived by left-concatenation of the prefix „un“ to the word „happy“**
- **„unhappy“ and „happy“ are two different words**

https://de.wikipedia.org/wiki/Wortbildung

- **Expresses grammatical functions of words in the sentence**
- **We can create the word „cats“ via inflection of the word „cat“ using the plural „-s“**
- **„cat“ and „cats“ are two forms of the same word**

https://de.wikipedia.org/wiki/Flexion

noun
verb
{affix}

- {prefix-} : „con-“ in „confirm“
- {-infix} : „bloody“ in „absobloodylutely“ – not present in German
- {-suffix} : „-ing“ in „studying“
- {circumfix} : „ex-“ and „-ed“ in „extended“

Interfix, duplifix, transflix, simulfix, supraflix, disfix, …

https://en.wikipedia.org/wiki/Affix
https://en.wiktionary.org/wiki/absobloodylutely

Morphology Tree: Example 1

Morphology

unbeliefable

https://www.youtube.com/watch?v=QT_A-7usieI&feature=emb_rel_end
https://all-about-linguistics.group.shef.ac.uk/branches-of-linguistics/morphology/what-is-morphology/

Morphology Tree: Example 1

Morph ology

Morphology

unbeliefable

Morphology Tree: Example 1

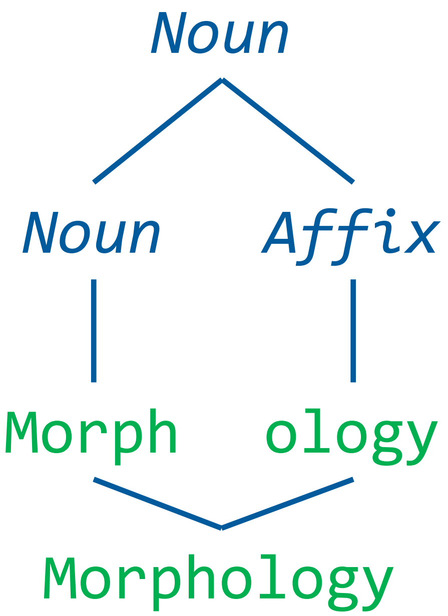
Noun *Affix*

Morph ology

Morphology

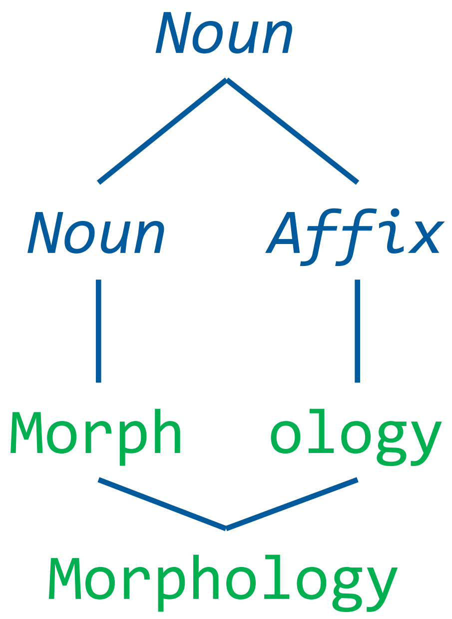
unbeliefable

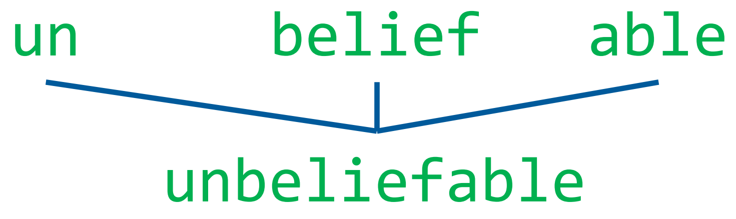
Morphology Tree: Example 1



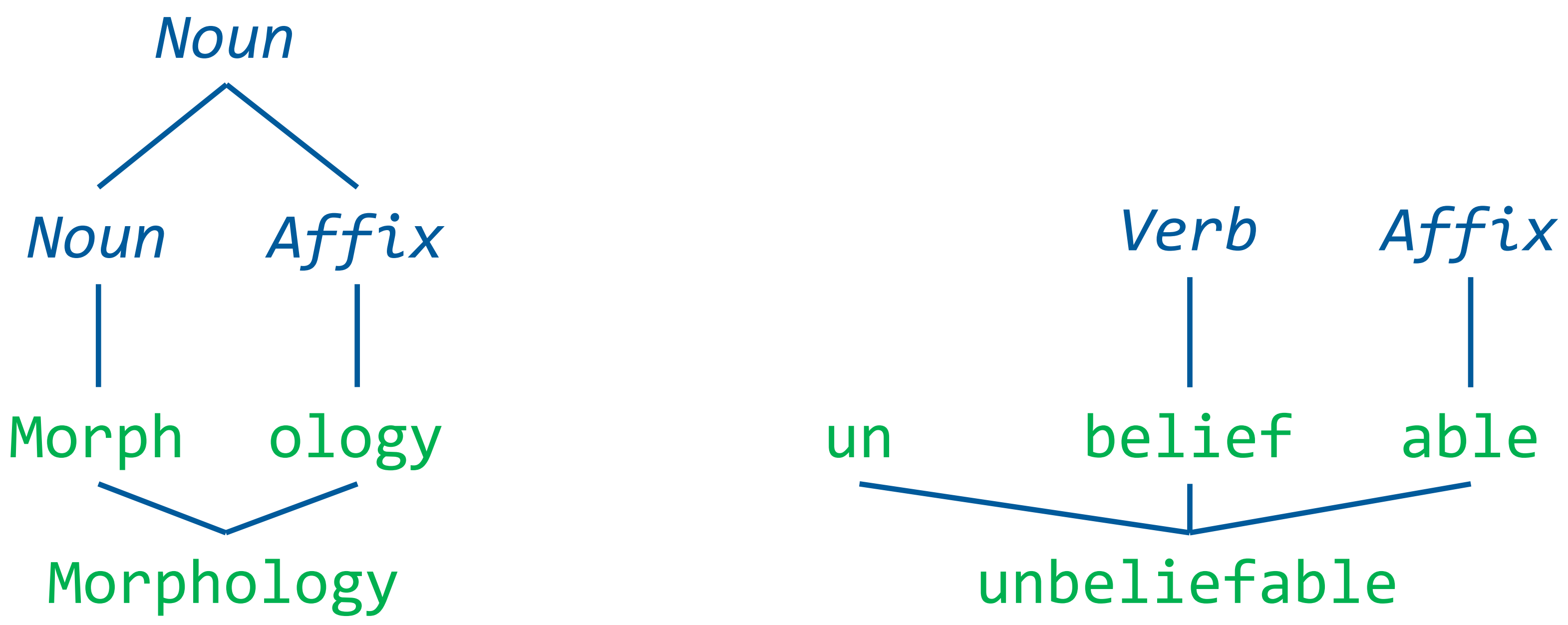
unbeliefable

Morphology Tree: Example 1

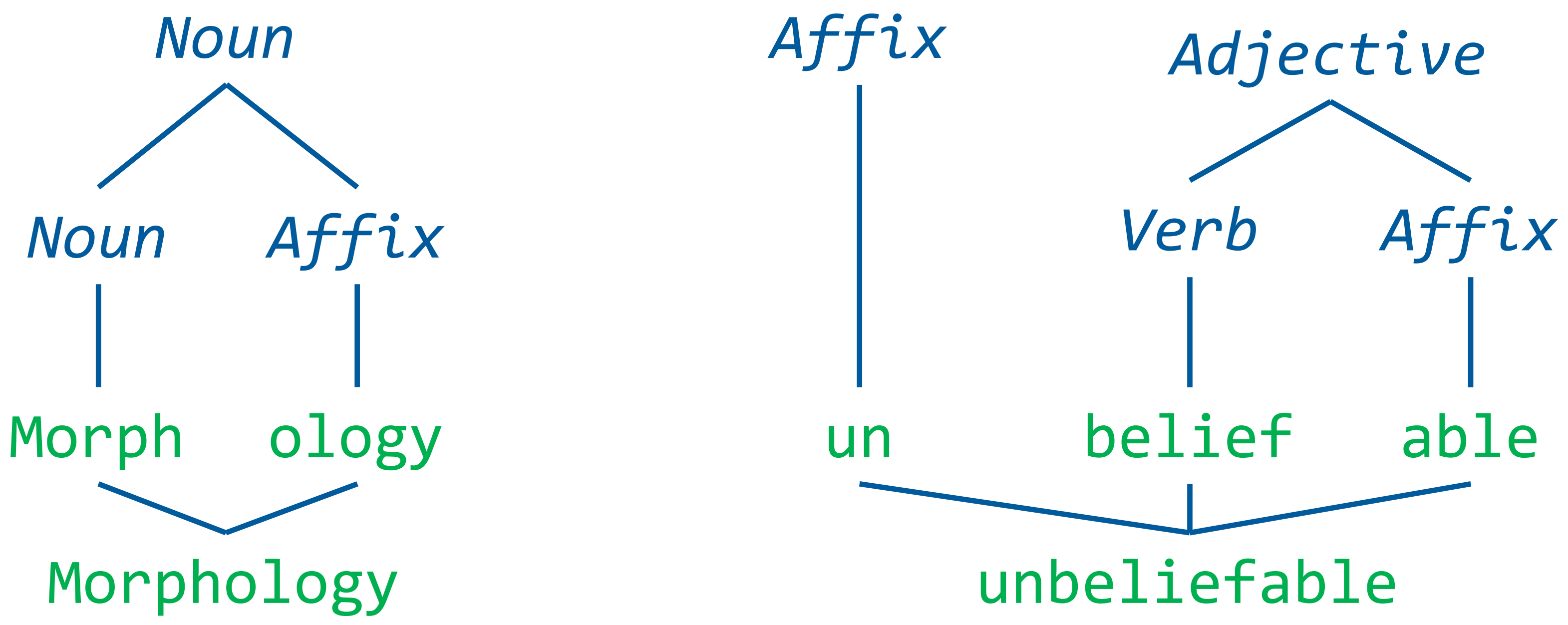




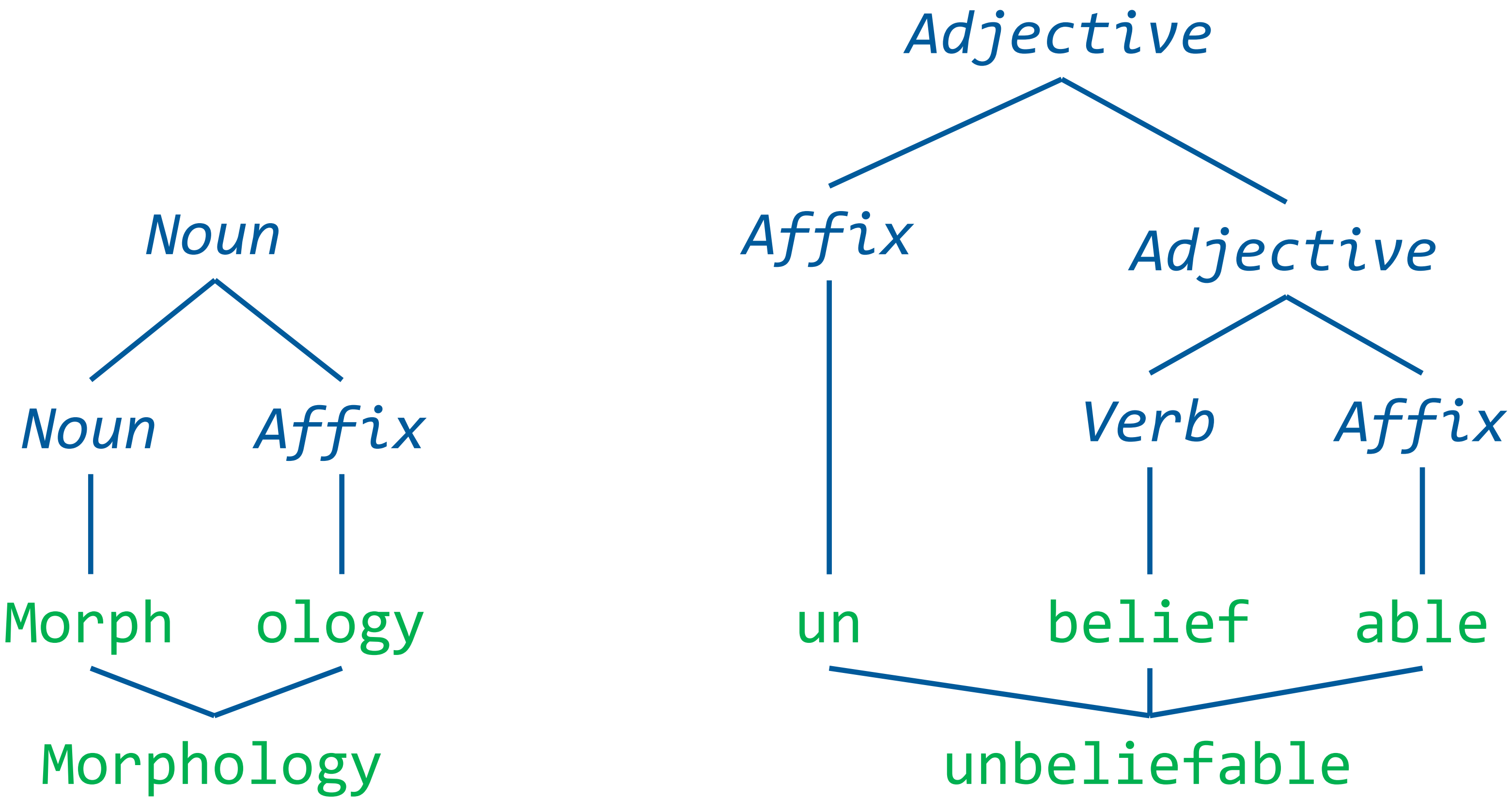
Morphology Tree: Example 1



Morphology Tree: Example 1



Morphology Tree: Example 1



Morphology Tree: Example 2

Antidisestablishmentarianism

Morphology Tree: Example 2

`Anti dis establish ment arian ism`

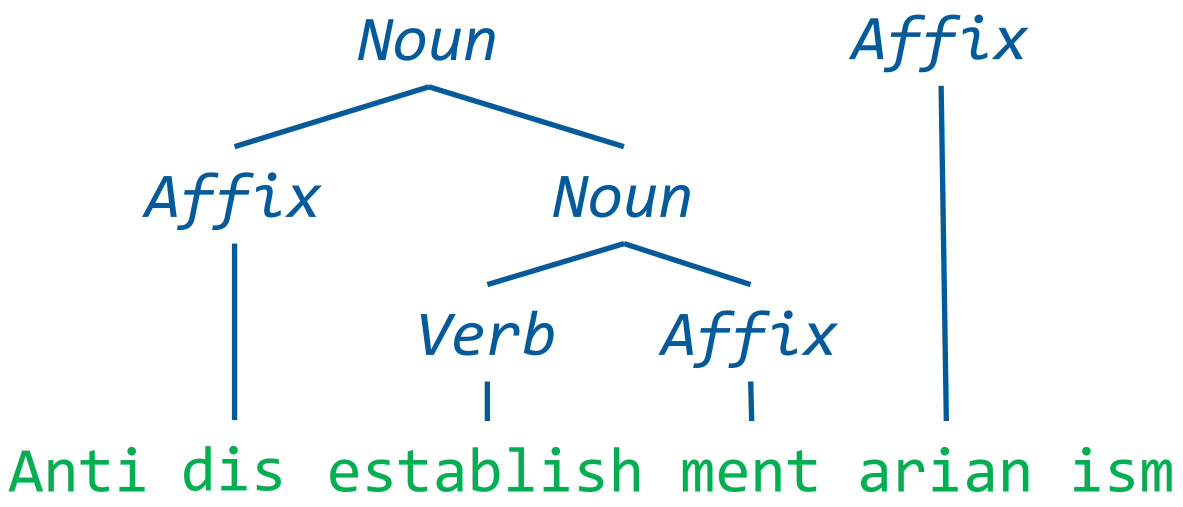
Morphology Tree: Example 2

```
              Verb    Affix
                |       |
Anti dis establish ment arian ism
```

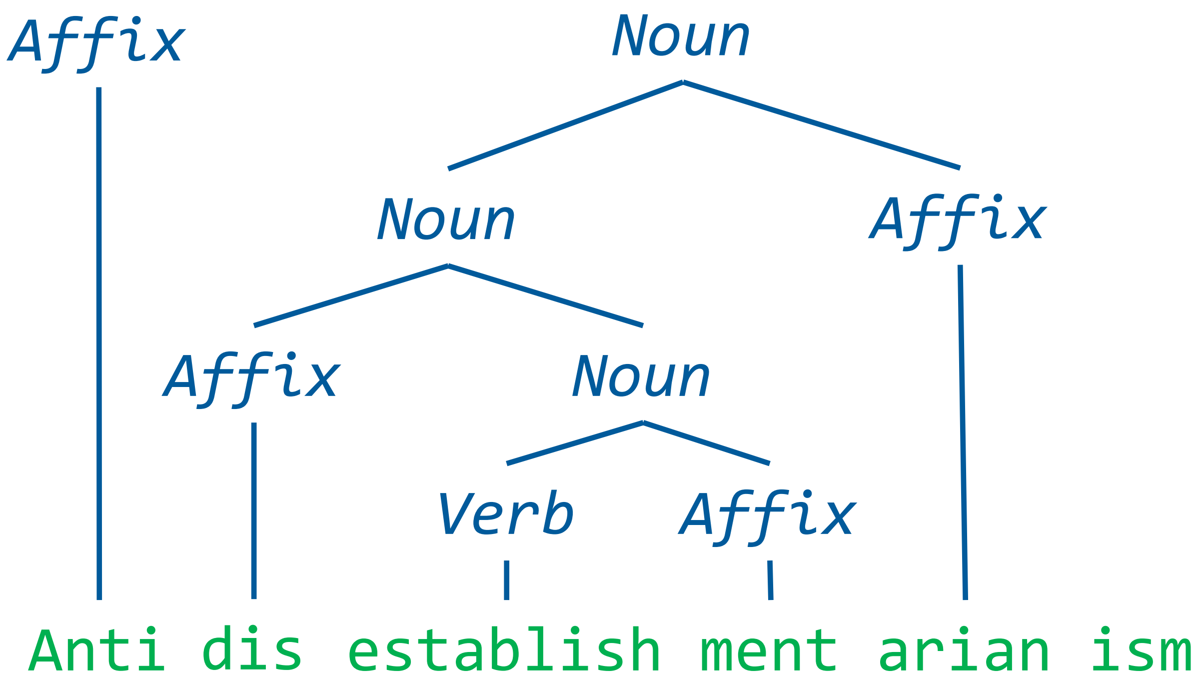
Morphology Tree: Example 2

```
     Affix        Noun
       |         /    \
       |      Verb    Affix
       |        |       |
Anti dis establish ment arian ism
```

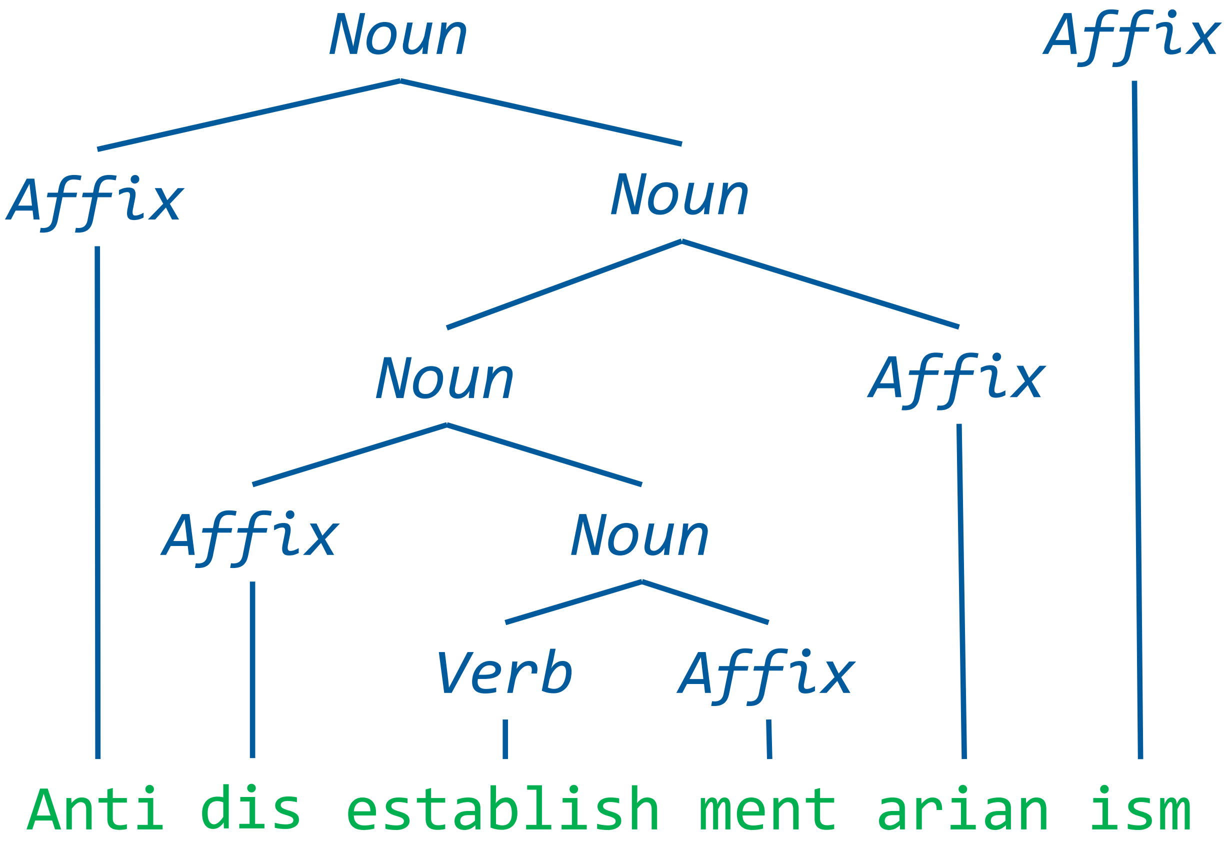
Morphology Tree: Example 2



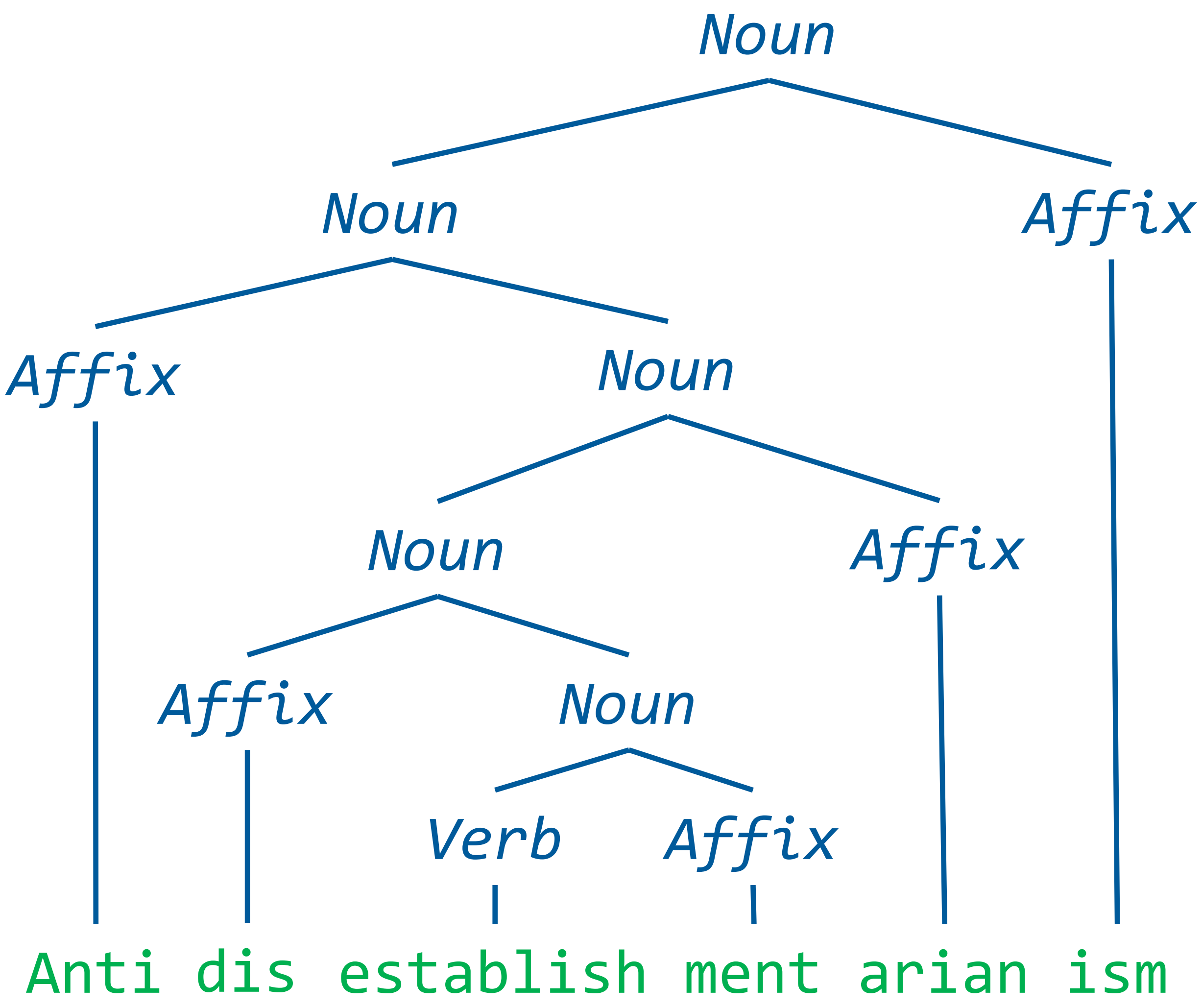
Morphology Tree: Example 2



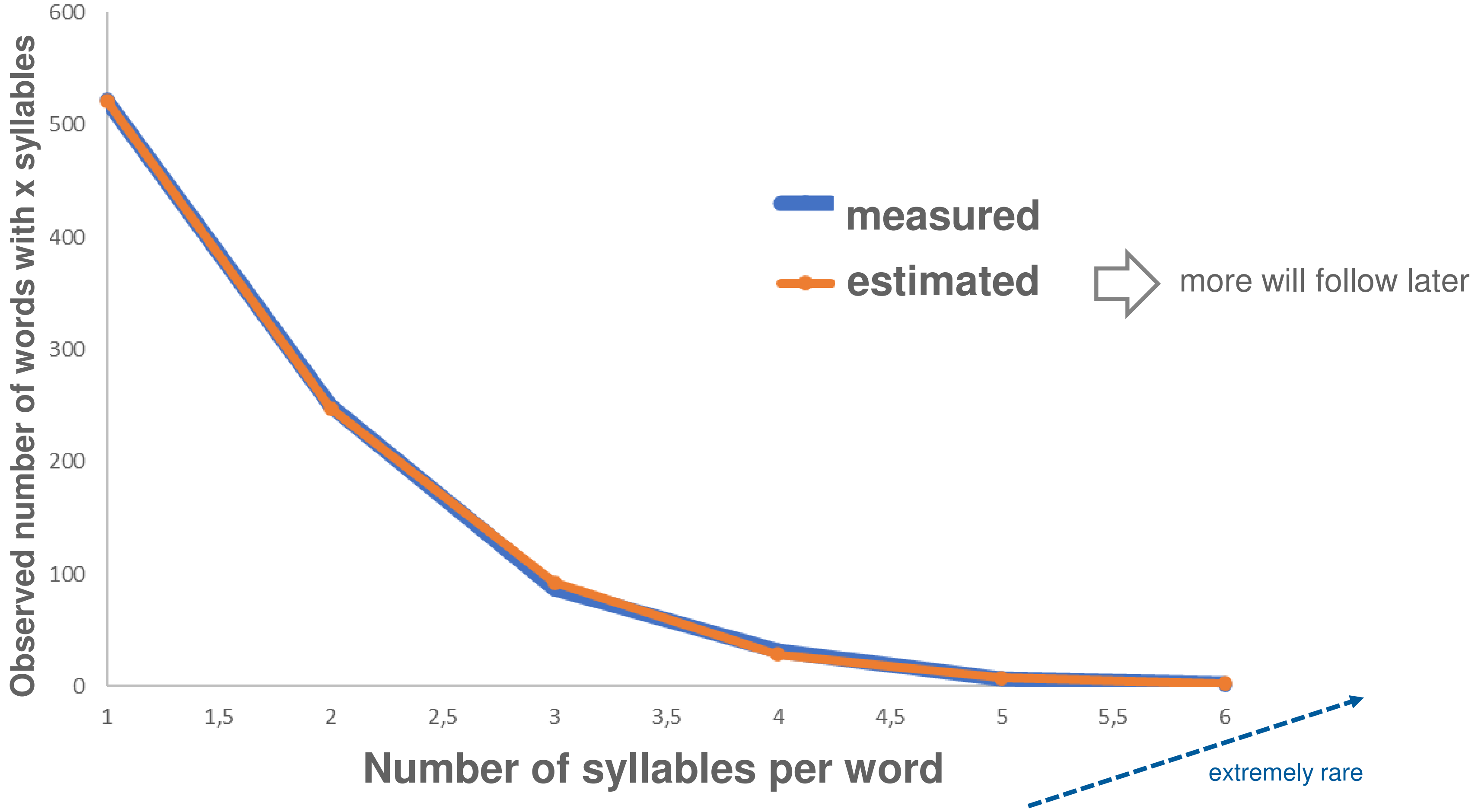
Morphology Tree: Example 2



Morphology Tree: Example 2



Word lengths



Lebensversicherungsgesellschaftsangesteller

See also: Towards a theory of word length distribution

Turkish

Example: “Uygarlastiramadiklarimizdanmissinizcasina”

(behaving) as if you are among those whom we could not civilize

Uygar	las	tir	ama	dik	lar	imiz	dan	mis	siniz	casina
Civilized	become	cause	not able	past	plural	p1pl	abl	past	2pl	as if

Do you know a better example?

Example: “legeslegmegszentségteleníttethetetlenebbjeitekként”

like the most of most undesecratable ones of you or as your most unsanctifiable

Do you know a better example?

https://github.com/oroszgy/awesome-hungarian-nlp#2-datasets

- Example cases of inflections:
  我(I) ->我们(we)
  他(he) ->他们 (them, plural)
  哥(friend) ->哥们(friends)

- **Adverbial adjective:**
  小心地做事 (do things carefully)

- **Adjective form of nouns:**
  可能 (can)
  可能性 (the possitility)

- **Adverbalized noun :**
  历史 (history)
  历史上 (in the history)

Lemmatization

Task of determining that two words have the same root, despite their surface differences

What is the basic form of the word?

Before Lemmatization	After Lemmatization
goose	goose
geese	goose
connects	connect
trouble	trouble
troubling	trouble
troubled	trouble
troubles	trouble

am, are, is, be, were, was => be
car, cars, car’s, cars’ => car

⇨ Complex rule-based systems

Stemming

Simpler version of lemmatization in which we mainly just strip suffixes from the end of the word

Porter’s algorithm

- **Martin Porter, 1980, An algorithm for suffix stripping, *Program*, 14(3) pp 130−137.**

„ trace related words to one and the same string“

- **Rule-based: https://tartarus.org/martin/PorterStemmer/def.txt**
- **Tony Kent Strix award in 2000**

Example

Input	Output
connect	connect
connected	connect
connections	connect
connects	connect
trouble	troubl
troubled	troubl
troubles	troubl
troublesome	troublesom

Stemming is crude chopping of affixes. It is language dependent
Example: automate(s), automatic – it is reduced to automat.

Porter’s algorithm

Example

forexample compressed and compression are both accepted as equivalent to compress

for ***exampl*** ***compress*** and ***compress*** ***ar*** both ***accept*** as ***equival*** to ***compress***

12 words

10 words

Possible Errors

Over-stemming or „false positive“
universal -> *univers*
university -> *univers*
universe -> *univers*
to „univers“

etymologically related but modern meanings are in widely different domains

These are not synonyms, search engine will likely reduce the relevance of the search results.

Stemming algorithms
To minimize both errors

Under-stemming or „false negative“
alumnus -> *alumnu*
alumni -> *alumni*
alumna/alumnae -> *alumna*

This English word keeps Latin morphology, and so these near-synonyms are not conflated.

Determining vocal-consonant-sequences

C := sequence of consonants
V := sequence of vocals
$(.)^m$:= m repetitions of "." with $m \geq 0$

$$[C](VC)^m[V]$$

```
tr ee
CC VV

 t o
 C V

  w  eb
  C (VC)¹

 an  t
(VC)¹ C

tr oubl e
CC VVCC V
 C (VC)¹ V

b etw een
C VCC VVC
C   (VC)²

tr oubl es
CC VVCC VC
 C   (VC)²

pr iv at e
CC VC VC V
 C (VC)² V

w ik ip ed ia
C VC VC VC VV
C   (VC)³   V
```

https://iq.opengenus.org/porter-stemmer/

Porter's algorithm

Shortening rules

(condition) S1 -> S2   if **<stem>S1** and **<stem>** satisfies **(condition)** then **<stem>S2**

1 of > 50 rules:

(m > 1) EMENT -> ‘’

```
<stem>S1 = REPLACEMENT
<stem>   = REPLAC
S1       = EMENT
```

m of <stem>:

REPLAC

C VCC VC

C (VC)^2

⇒ m=2 > 1

Shorten with (m > 1) EMENT ->‘‘

⇒ REPLACEMENT wird **REPLAC**

Porter’s algorithm

Shortening rules

(condition) S1 -> S2 if **<stem>S1** and **<stem>** satisfies **(condition)** then **<stem>S2**

Example conditions:

*S - the stem ends with S (and similarly for the other letters).
v - the stem contains a vowel.
m=2 TROUBLES, PRIVATE, OATEN, ORRERY.
*d - the stem ends with a double consonant (e.g. -TT, -SS).
*o - the stem ends cvc, where the second c is not W, X or Y (e.g. -WIL, -HOP).

Stemming vs. Lemmatization

- **Stemming always shortens the word!**
- **When we apply lemmatization, the word stem does not even need to be the same: (to be, is, was, were)**

Stemming is used most often.