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!“

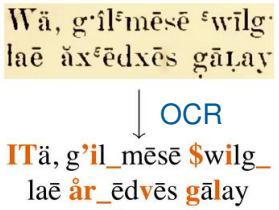
- „uh“: should we also count speech disfluencies?
- „main-“ How to count fragments?
- What about plural –s?

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

Gierafe

Correct spelling with 7 characters

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

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

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

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

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 $d(x, y) = d(y, x)$ 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** $$WER = \frac{S + D + I}{N}$$

Levenshtein-Distance

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-1, j] + 1                          // deletion
d[i, j]:= min                     ⎪ d[i, j-1] + 1                          // insertion
                                  ⎨ d[i-1, j-1] + ⎧ 2; if X[i] ≠ Y[j]      // substitution
                                  ⎩               ⎩ 0; if X[i] = Y[j]
```

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“

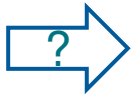
#Tokens: 5

#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

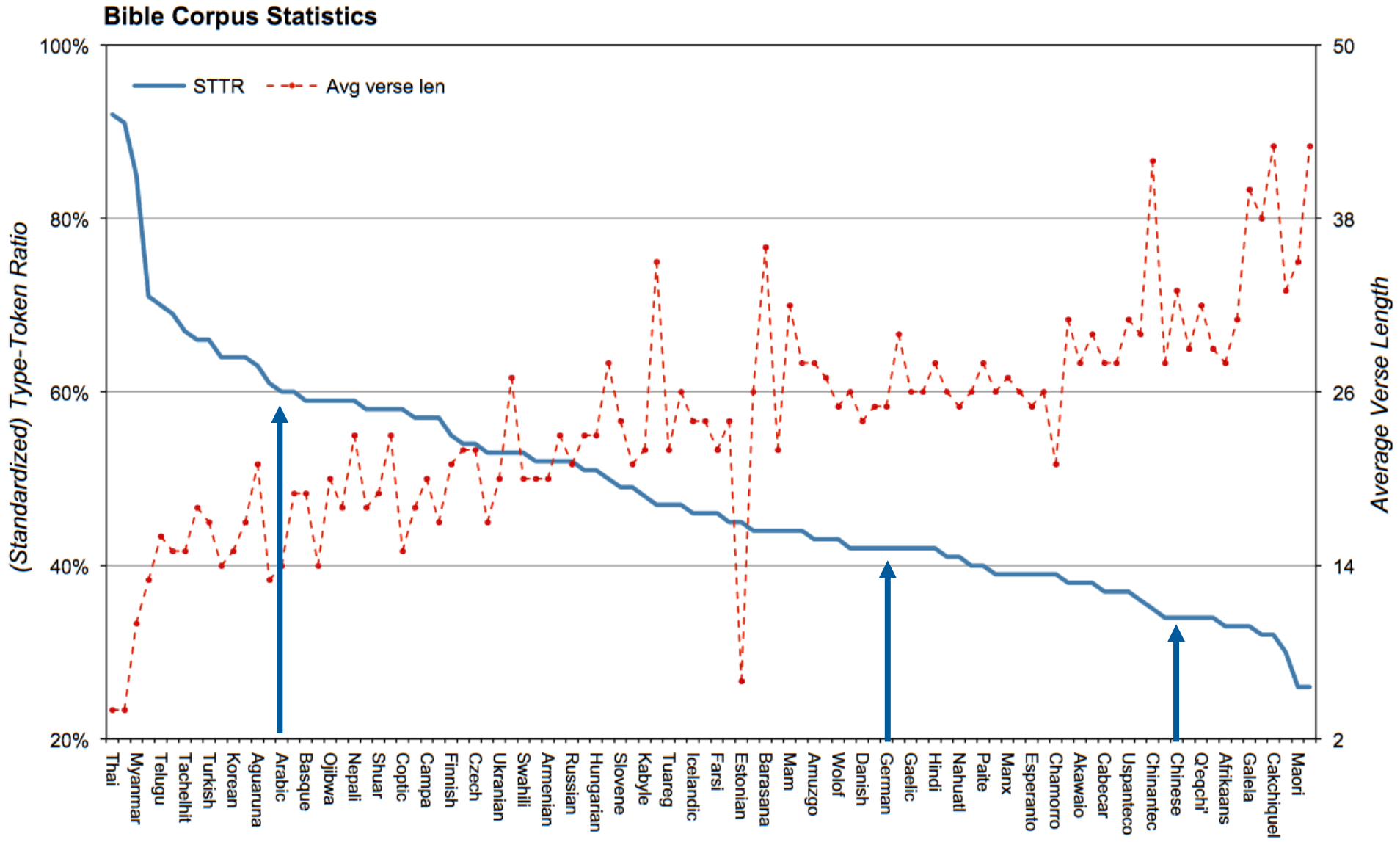
- **Church & Gale (1990): |Typen| > O(|Tokens|$^{0.5}$)**

	|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

Words in other Languages

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

Words in other Languages

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

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? – 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

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

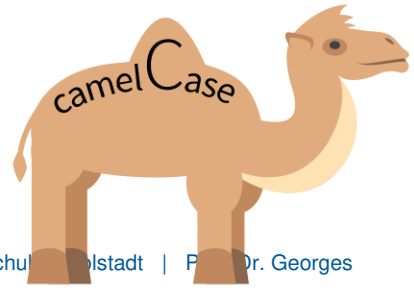
- **[!”#$%&’()*+,-./:;<=>?@[\]^_`{|}~]**
- **Space, line break**
- **<tr>, <a>, <p>, …**

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