

Arbeiten mit Textdaten

Simon Schölzel, M.Sc.

(updated: 11.05.2023)

1

Einführung und Motivation

2

Regular Expression

3

Natural Language Processing

1

Einführung und Motivation

2

Regular Expression

3

Natural Language Processing

In den Wirtschaftswissenschaften sind wir erprobt im Umgang mit **strukturierten, tabellarischen Daten**, insbesondere dem finanziellen Zahlenwerk, das die Waren- und Geldströme in der Wirtschaft abbildet.

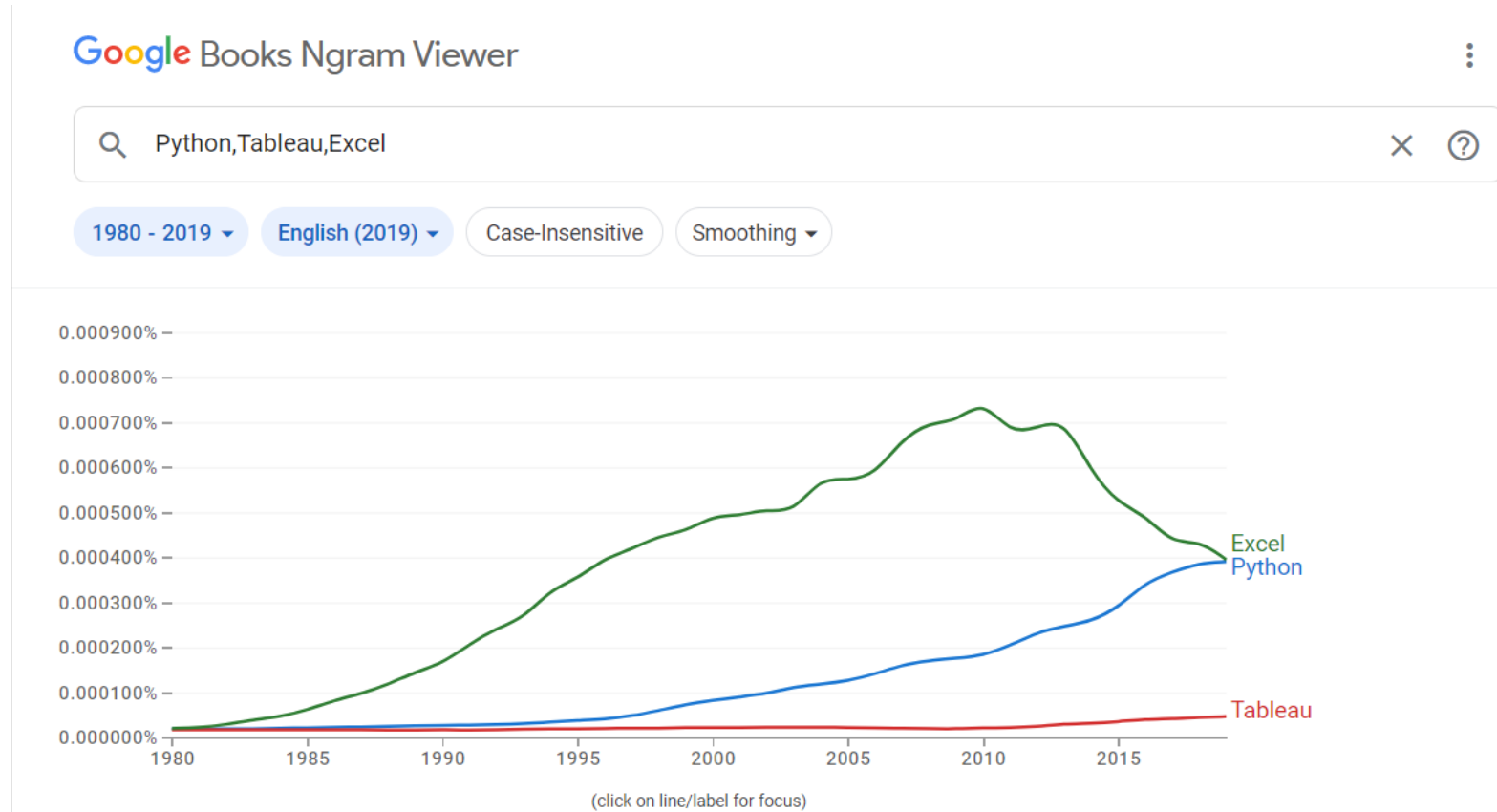
Ein signifikanter Teil der täglich entstehenden Daten liegt in **unstrukturierter Form** vor und ist Ausdruck wirtschaftlichen Handelns sowie menschlicher Interaktion, Kommunikation und kultureller Phänomene.

Einige Anwendungsbeispiele für Textdaten

- » **Finance:** Verwendung von Finanz-News, Social Media (z.B. Twitter, <https://seekingalpha.com/>, r/wallstreetbets), oder Unternehmensberichten zur Prognose von Gewinnen, Insolvenzwahrscheinlichkeiten oder Bilanzmanipulationen.
- » **Marketing:** Analyse der Inhalte von Online Werbung und Produkt Rezensionen und deren Einfluss auf das Entscheidungsverhalten von Konsumenten.
- » **Volkswirtschaftslehre:** Vorhersage von Inflationserwartungen, Schwankungen der Arbeitslosenrate oder politischer Unsicherheit anhand von News-Daten.
- » **Politik:** Untersuchung der Treiber und Effekte politischer Einstellungen von Bürgern anhand von Social Media Posts und Profilen sowie der Dynamik politischer Debatten anhand von politischer Reden.
- » **Produktentwicklung:** Training von Machine Learning Modellen anhand von Textdaten zur automatisierten Übersetzung, Transkription, Zusammenfassung von Texten, Textgenerierung oder Einsatz als Chatbots.

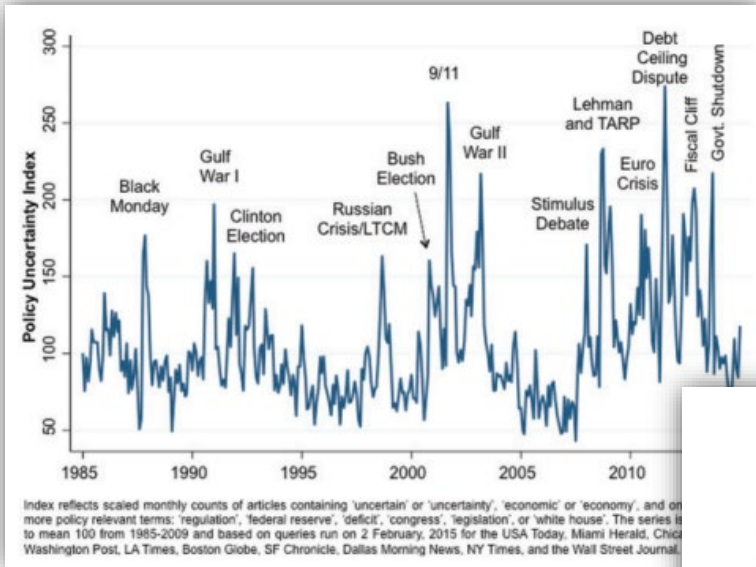
1 Einführung und Motivation

1.1 Relevanz von Textdaten

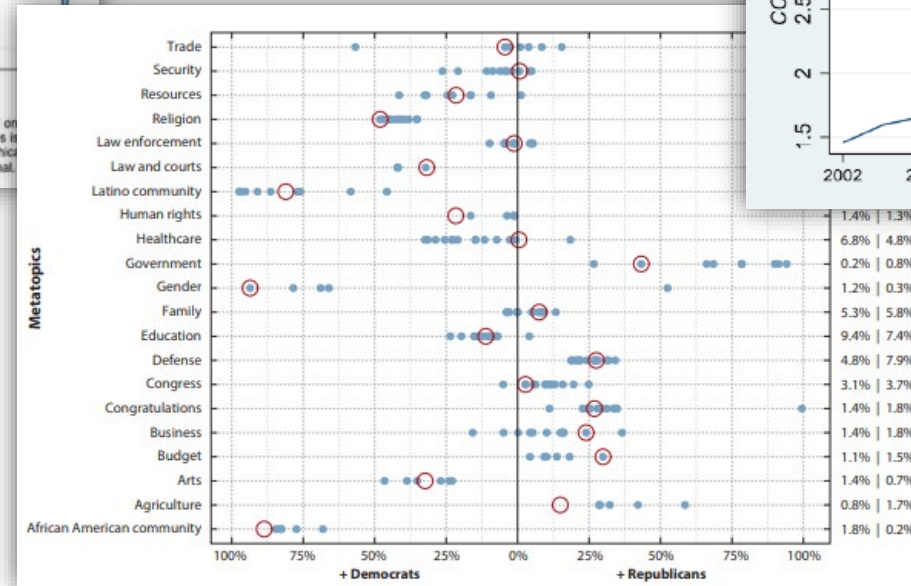


1 Einführung und Motivation

1.1 Relevanz von Textdaten



Quelle: [Baker/Bloom/Davis \(2016\)](#)



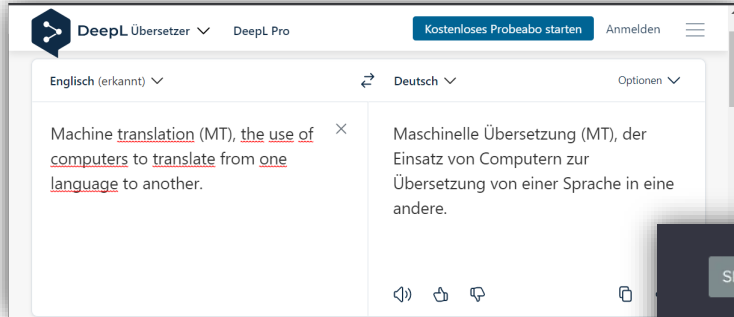
Quelle: [Wilkerson/Casas \(2017\)](#)



Quelle: [Sautner et al. \(2022\)](#)

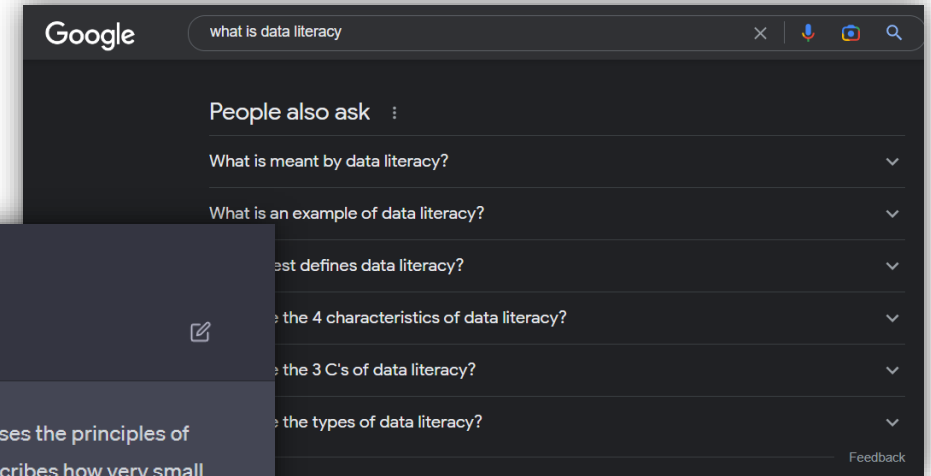
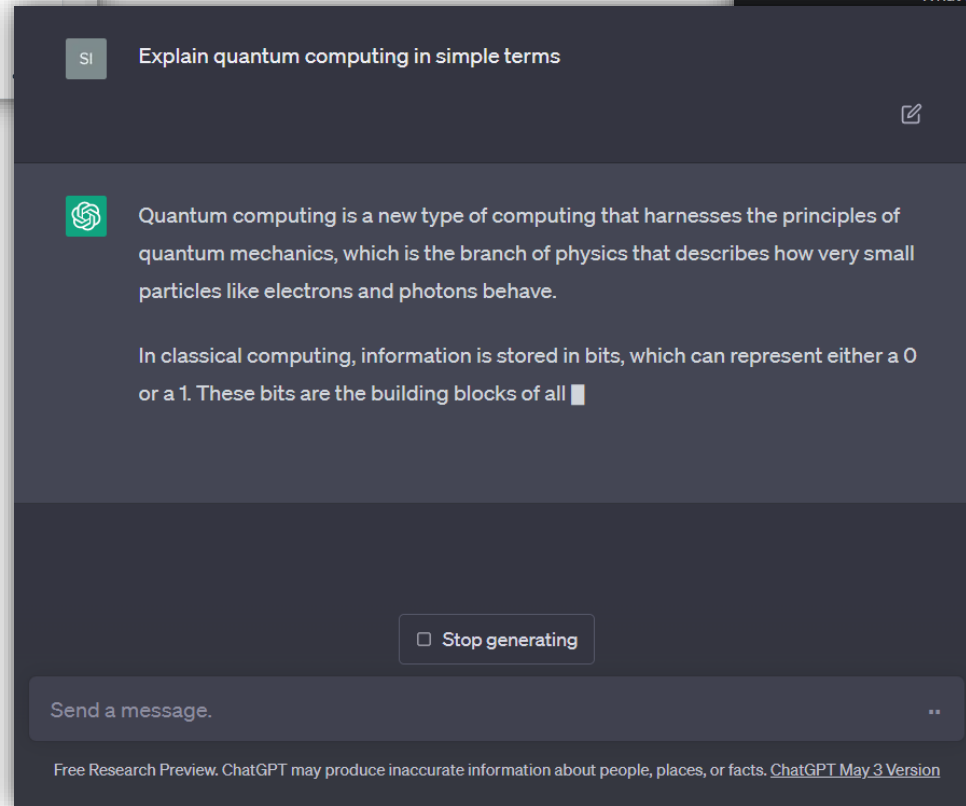
1 Einführung und Motivation

1.1 Relevanz von Textdaten



Machine Translation

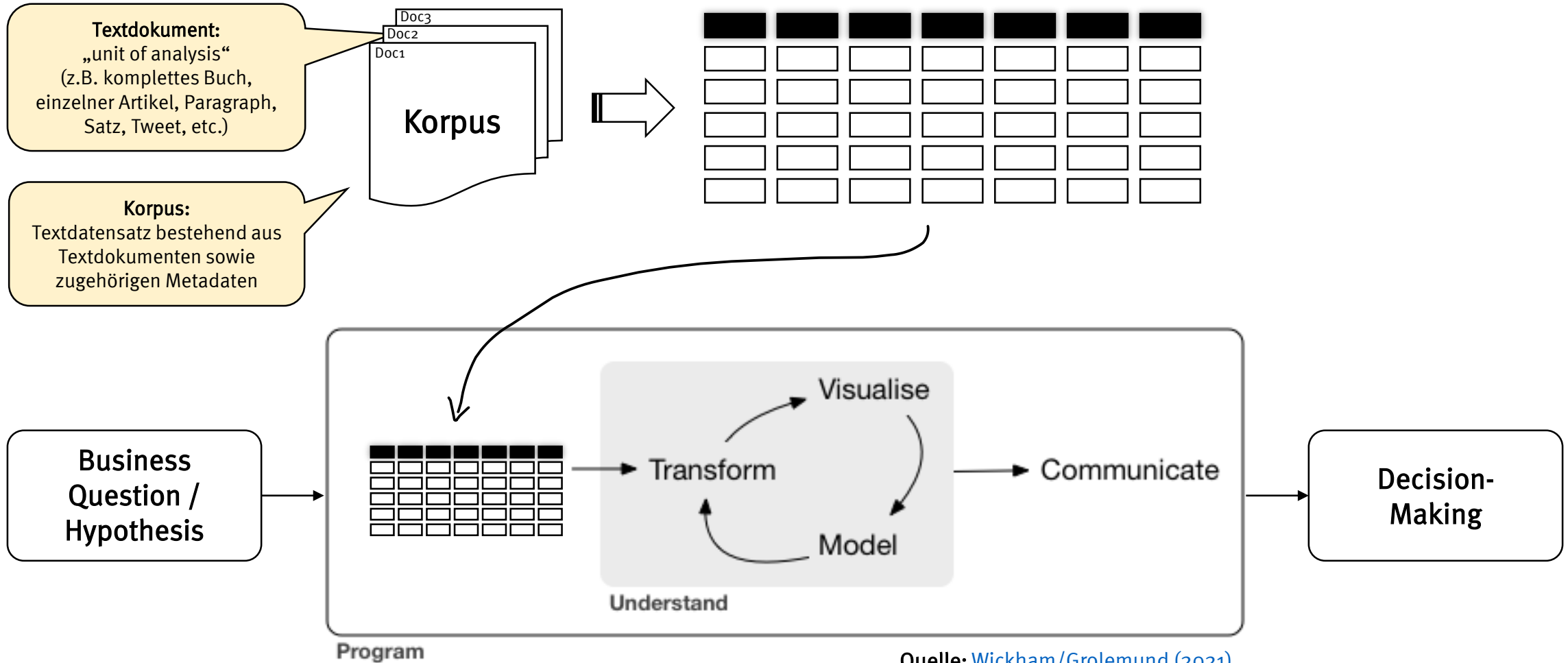
Chatbots and Dialogue Systems



*Question Answering and
Information Retrieval*

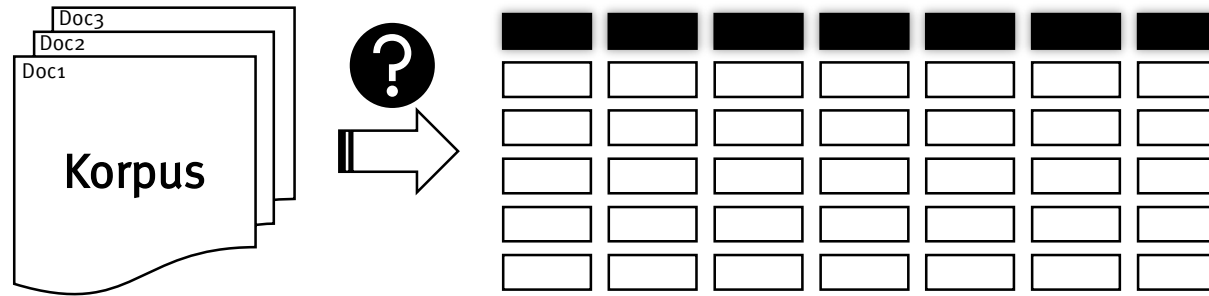
1 Einführung und Motivation

1.2 Datenanalyse mittels Textdaten



1 Einführung und Motivation

1.2 Datenanalyse mittels Textdaten



1) TEXT PREPROCESSING:

- » Bereinigung der rohen Texte von unerwünschten Artefakten (z.B. Satzzeichen, HTML-Fragmente, Überschriften, Personen-Namen, Füllwörtern, etc.) unter Verwendung von Regular Expression (REGEX) oder dedizierten Textanalyse Bibliotheken (z.B. `spacy`, `nltk`, `textblob`).
- » Segmentierung von Texten, z.B. splitten in Paragraphen oder einzelne Wörter (TOKENIZATION).
- » Normalisierung von Wörtern (STEMMING / LEMMATIZATION).

2) Transformation: Umwandlung der bereinigten, unstrukturierten Textdaten in ein strukturiertes, tabellarisches Format (TERM-DOCUMENT-MATRIX).

1

Einführung und Motivation

2

Regular Expression

3

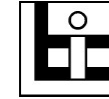
Natural Language Processing

2 Regular Expression

2.1 Einführung

String:

Beliebige Folge an Zeichen, die einen Text ergeben (siehe Python-Vorlesung)



Forschungsteam
Berens

- » **Regular Expressions (REGEX)** sind Zeichenfolgen, die mittels eigener Syntax Muster in einem `string` identifizieren („*Wildcards on Steroids*“). Wir nennen dieses Vorgehen auch **STRING MATCHING**.
- » Häufig auftretende Muster in Textdaten sind z.B. Satzzeichen zur Identifizierung von abgeschlossenen Sätzen, E-Mail Adresse, URLs, Zitate, positive/negative Wörter (Sentiment Analysis) oder Namen.
- » In Python können wir regex über das `re` Modul nutzen.

```
import re
```

```
string = "Python (['pʰaɪθn], ['pʰaɪθɒn], auf Deutsch auch ['pʰy:tɒn]) ist eine  
universelle, üblicherweise interpretierte, höhere Programmiersprache.[12] Sie hat den  
Anspruch, einen gut lesbaren, knappen Programmierstil zu fördern.[13] So werden  
beispielsweise Blöcke nicht durch geschweifte Klammern, sondern durch Einrückungen  
strukturiert."
```

Regular Expression (Suchmuster)

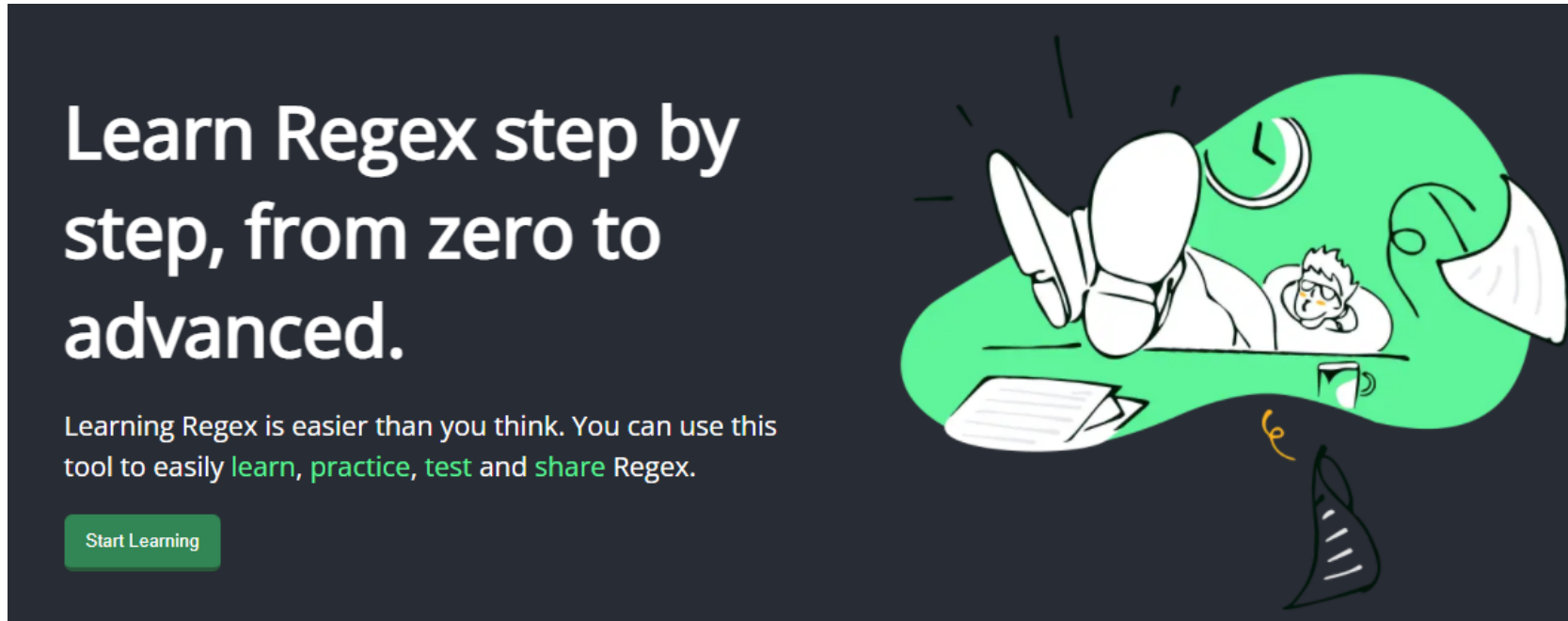


```
re.findall(r'\[.*?\]', string)
```

```
> ['['pʰaɪθn]', '['pʰaɪθɒn]', '['pʰy:tɒn]', '[12]', '[13]]'
```

2 Regular Expression

2.2 Praktische Übung



Learn Regex step by step, from zero to advanced.

Learning Regex is easier than you think. You can use this tool to easily **learn**, **practice**, **test** and **share** Regex.

[Start Learning](#)

Die folgenden Beispiele stammen von der Lernplattform

<https://regexlearn.com/>

2 Regular Expression

2.3 Suchbefehle und grundlegende Metacharacters

Suchbefehle:

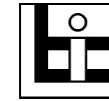
- » `re.match(regex, string)`: Suche nach `regex` am Anfang des `string`
- » `re.search(regex, string)`: Suche nach `regex` im gesamten `string`
- » `re.search(regex, string, flags=re.IGNORECASE)`: Case-insensitive Suche
- » `re.findall(regex, string)`: Suche und Extrahiere alle `regex` in `string`

Basic Metacharacters:

- » `.`: Beliebiges Zeichen (inkl. Leerzeichen)
- » `[a-z]`: Zeichenklasse (hier: Buchstabe a, b, c, ..., y, z) \mapsto Kann wie ein „oder“ verstanden werden
- » `[a-z0-9]`: Zeichenklasse (hier: Buchstabe a, b, c, ..., y, z oder Zahl 0, 1, 2, ..., 8, 9)
- » `[^m-z]`: Komplement, d.h. Zeichen, die nicht in der Zeichenklasse aufgeführt sind

2 Regular Expression

2.3 Suchbefehle und Suchmuster: Übung



Übungsaufgaben:
10 Minuten



2 Regular Expression

2.4 Weitere Metacharacters

Escape Character: Der Backslash wird verwendet, um Metacharacters zu „umgehen“ (ESCAPING)

- » “\.” matcht “.”
- » “\[\]” matcht “[]”

Wiederholungen:

- » * : Keinalmal, einmal oder mehrmals
- » + : Einmal oder mehrmals
- » ? : Keinalmal oder einmal (optional)
- » { 2 } : Genau zweimal
- » { 2, } : Zweimal oder häufiger
- » { 2, 4 } : Zweimal, dreimal oder viermal

Anchors:

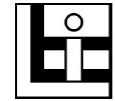
- » ^ : Beginn des Textes (Achtung: innerhalb einer Zeichenklasse “[]” als Komplement interpretiert!)
- » \$: Ende des Textes



Quelle: [RexEgg](https://www.rexegg.com/) Regex Cheat Sheet

2 Regular Expression

2.4 Weitere Metacharacters: Übung



Forschungsteam
Berens



Übungsaufgaben:
10 Minuten



Gruppen und Alternations:

- » `()` : Suche nach einer Gruppe von Zeichen, die gemeinsam ein Muster abbilden
- » `|` : Suche nach alternativen Zeichenketten \mapsto Kann wie ein „oder“ auf Wort-Level verstanden werden

Andere spezielle Zeichen:

- » `\w` : Suche nach Buchstaben, Zahlen und Unterstrichen (analog zu `[a-zA-Z0-9_]`)
- » `\d` : Suche nach Zahlen
- » `\s` : Suche nach Leerzeichen (z.B. einfaches Leerzeichen, Tabstop oder Newline)

2 Regular Expression

2.5 Regex in der Praxis



Quelle: [tenor](#)



» **Praxistipp:** Versuch durch ein iteratives Vorgehen mit diversen Validierungsschritten das Risiko für Typ I and Typ II Fehler minimieren.

1. **Typ I Fehler:** FALSE POSITIVES (matchen von strings, die nicht gematcht werden sollen)
2. **Typ II Fehler:** FALSE NEGATIVES (nicht matchen von strings, die gematcht werden sollen)

2 Regular Expression

2.6 Anwendungsfälle: Name Matching

- » Häufig stehen wir vor dem Problem, dass wir Informationen aus verschiedenen Datensätzen verbinden möchten (JOINEN/MERGEN). Je nach Ursprung des Datensatzes, unterscheiden sich die Benennungen von Unternehmen, Personen, Orten, etc.
- » Regular Expressions ermöglichen es, Strings zu normalisieren, sodass ein Join möglich wird.

```
coname1 = 'SS&C TECHNOLOGIES HOLDINGS INCORPORATED'  
coname2 = 'SS&C Technologies Hldgs Inc.'
```

|
Regex

```
coname1 ↓ 'ssc technologies holdings'  
coname2 = 'ssc technologies holdings'
```

```
coname1 == coname2
```

2 Regular Expression

2.6 Anwendungsfälle : Name Matching

coname1	coname1_norm	coname2_norm	coname2	similarity
INTEGRA LIFESCIENCES HOLDINGS CORP	integralifesciencesholdings	integralifesciencesholdngs	INTEGRA LIFESCIENCES HOLDNGS	98
VENTANA MEDICAL SYSTEMS INC	ventanamedicalsystems	ventanamedicalsystem	VENTANA MEDICAL SYSTEM INC	98
CORRECTIONS CORP OF AMERICA MD	correctionscorpofamericamd	correctionscorpofamerica	CORRECTIONS CORP OF AMERICA	96
SCIELE PHAMA INC	scielephama	scielepharma	SCIELE PHARMA INC	96
BRADLEY PHARMACEUTICALS INC	bradleypharmaceuticals	bradleypharmaceuticl	BRADLEY PHARMACEUTICL -CL A	95
INTERACTIVE INTELLIGENCE INC	interactiveintelligence	interactiveintelligencegrp	INTERACTIVE INTELLIGENCE GRP	94
TAKE TWO INTERACTIVE SOFTWARE INC	taketwointeractivesoftware	taketwointeractivesftwr	TAKE-TWO INTERACTIVE SFTWR	94
ZEBRA TECHNOLOGIES CORP	zebratechnologies	zebratechnologiescp	ZEBRA TECHNOLOGIES CP -CL A	94
LINCOLN ELECTRIC HOLDINGS INC	lincolnelectricholdings	lincolnelectrichldgs	LINCOLN ELECTRIC HLDGS INC	93
SS&C TECHNOLOGIES INC	ssctechnologies	spstechnologies	SPS TECHNOLOGIES INC	93
NATURE S SUNSHINE PRODUCTS INC	naturessunshineproducts	naturessunshineprods	NATURES SUNSHINE PRODS INC	93
SS&C TECHNOLOGIES HOLDINGS INC	ssctechnologiesholdings	ssctechnologieshldgs	SS&C TECHNOLOGIES HLDGS INC	93
BROCADE COMMUNICATIONS SYSTEMS INC	brocadecommunicationssystems	brocadecommunicationssys	BROCADE COMMUNICATIONS SYS	92
PALOMAR MEDICAL TECHNOLOGIES INC	palomarmedicaltechnologies	palomarmedtechnologies	PALOMAR MED TECHNOLOGIES INC	92
SONIC AUTOMOTIVE INC	sonicautomotive	sonicautomotiveinc	SONIC AUTOMOTIVE INC -CL A	91
STARWOOD HOTELS & RESORTS WORLDWIDE INC	starwoodhotelsresortsworldwide	starwoodhotelsresortswrld	STARWOOD HOTELS&RESORTS WRLD	91

1

Einführung und Motivation

2

Regular Expression

3

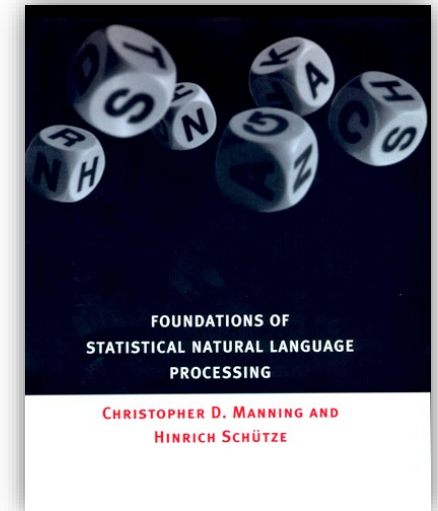
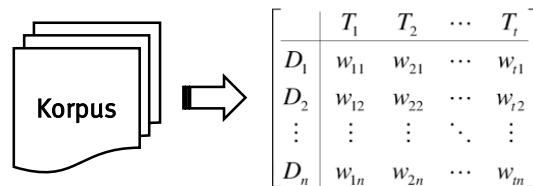
Natural Language Processing

3 Natural Language Processing

3.1 Überblick

Natural Language Processing (NLP): Konzepte und Methoden zur automatisierten und computer-gestützten Verarbeitung natürlicher Sprache in Texten.

- » **Tokenization:** Unterteilung eines Textdokuments in inhaltlich bedeutsame Einheiten (TOKEN), z.B. einzelne Wörter, N-GRAMS, Sätze oder Absätze.
- » **Stop Word Removal:** Entfernen von inhaltlosen Füllwörtern.
- » **Stemming / Lemmatization:** Normalisierung von Wörtern durch Reduktion auf die Stammform, z.B. „gehen“, „geht“, „ging“, „gegangen“ \mapsto „gehen“.
- » **Transformation:** Überführung in tabellarisches Format (Document-Term-Matrix)
 - » Zeilen bilden einzelne Dokumente ab
 - » Spalten bilden einzelne Token (z.B. Wörter) ab
 - » Die Einträge der Matrix bilden Worthäufigkeiten ab



spaCy

NLTK

3 Natural Language Processing


3.2 Tokenization

„Sure, Alan. This quarter, because of the success we had with the bottom line, that ties directly into the way that we're compensating our management team, which is bottom line profitability. So the variable compensation expense is what drove the majority of the corp G&A increase. And then we also -- this quarter, there were some shares that were issued as well so that there was some share-based compensation expense that hit as well.“

z.B. REGEX-basiert
(`re.split(' ', string)`)

(space-based)

Tokenization

- » Interpunktion (z.B. Ph.D., M&A, 3.50€, 2022-06-03, <https://www.wiwi.uni-muenster.de/>, #data)
- » Zusammengesetzte Wörter (z.B. internet-of-things, Lebensversicherungsgesellschaftsangestellte) 
- » Sprachen, die ohne Leerzeichen auskommen (z.B. Chinesisch)

Sure, Alan, This, quarter, because, of, the, success, we, had, with, the, bottom, line, that, ties, directly, into, the, way, that, we, re, compensating, our, management, team, which, is, bottom, line, profitability, So, the, variable, compensation, expense, is, what, drove, the, majority, of, the, corp, G&A, increase, And, then, we, also, this, quarter, there, were, some, shares, that, were, issued, as, well, so, that, there, was, some, share, based, compensation, expense, that, hit, as, well

Sure, Alan, This, quarter, because, of, the, success, we, had, with, the, bottom, line, that, ties, directly, into, the, way, that, we, re, compensating, our, management, team, which, is, bottom, line, profitability, So, the, variable, compensation, expense, is, what, drove, the, majority, of, the, corp, G&A, increase, And, then, we, also, this, quarter, there, were, some, shares, that, were, issued, as, well, so, that, there, was, some, share, based, compensation, expense, that, hit, as, well

Stop Word Removal

- » Stop Word Removal ist eine Methode zur Reduktion der Dimensionalität der Daten
- » Stop Word Removal liegt eine Annahme über den Informationsgehalt bestimmter Wörter zugrunde
- » Je nach Anwendungsfall können Füllwörter sehr informativ sein (z.B. Ähnlichkeit von Sprechern, „Schwafelei“/Ambiguity)

Sure, Alan, This, quarter, because, of, the, success, we, had, with, the, bottom, line, that, ties, directly, into, the, way, that, we, re, compensating, our, management, team, which, is, bottom, line, profitability, So, the, variable, compensation, expense, is, what, drove, the, majority, of, the, corp, G&A, increase, And, then, we, also, this, quarter, there, were, some, shares, that, were, issued, as, well, so, that, there, was, some, share, based, compensation, expense, that, hit, as, well

3 Natural Language Processing

3.4 Lemmatization

Sure, Alan, quarter, success, bottom, line, ties, directly, way, compensating, management, team, bottom, line, profitability, variable, compensation, expense, drove, majority, corp, G&A, increase, also, quarter, share, issued, well, share, based, compensation, expense, hit, well

a) Lemmatization:

Reduktion auf Lemma (hier)

compensating -> compensate



Normalization

b) Stemming:

Reduktion auf Wortstamm (alt.)

compensating -> compens

Quelle: [Dan Jurafsky](#)

	Tokens = N	Types = V
Switchboard phone conversations	2.4 million	20 thousand
Shakespeare	884,000	31 thousand
Google N-grams	1 trillion	13 million

- » Weitere Reduktion der Dimensionalität der Daten
- » Verhinderung der Einzelzählung gleicher Wörter

sure, alan, quarter, success, bottom, line, **tie**, directly, way, **compensate**, management, team, bottom, line, profitability, variable, compensation, expense, **drive**, majority, corp, g&a, increase, also, quarter, share, **issue**, well, share, **base**, compensation, expense, hit, well

3 Natural Language Processing

3.5 Named Entity Recognition und Collocation Modeling

sure, alan, quarter, success, bottom, line, tie, directly, way, compensate, management, team, bottom, line, profitability, variable, compensation, expense, drive, majority, corp, g&a, increase, also, quarter, share, issue, well, share, base, compensation, expense, hit, well

**Named Entity
Recognition (NER)**

"You shall know a word by the company it keeps!"
~John Rupert Firth (1957)

**Collocation
Modeling**

n-gram:
Sequenz benachbarter
Token (manuell definiert
oder statistisch signifikant
auftretend)

sure, [PERSON], [DATE], success, bottom_line, tie, directly, way, compensate, management_team, bottom_line, profitability, variable_compensation, expense, drive, majority, corp, [ORG], increase, also, [DATE], share, issue, well, share_base_compensation, expense, hit, well

3 Natural Language Processing

3.6 Document-Term-Matrix

sure, [PERSON], [DATE], success, bottom_line, tie, directly, way, compensate, management_team, bottom_line, profitability, variable_compensation, expense, drive, majority, corp, [ORG], increase, also, [DATE], share, issue, well, share_base_compensation, expense, hit, well

Document-Term-Matrix (DTM)

	also	bottom_line	compen_sate	corp	directly	drive	expense	profitab_ility	[DATE]
Doc 1	1	2	1	1	1	1	1	1	1
Doc 2	0	2	1	0	2	0	0	3	0
...	1	3	0	0	0	1	1	1	4
Doc n-1	1	2	4	0	0	0	2	2	1
Doc n	0	0	1	0	1	0	0	0	0

Document Vector:
Numerische
Representation des
Textes (z.B. anhand
der Anzahl der Token)

In tidy data:

- each variable forms a column
- each observation forms a row
- each cell is a single measurement

Measurement / Gewichtung:
absolute Häufigkeit, relative
Häufigkeit, binäre Häufigkeit,
tf-idf, etc.

3 Natural Language Processing

3.7 Bag-of-Words (BoW) Assumption

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!



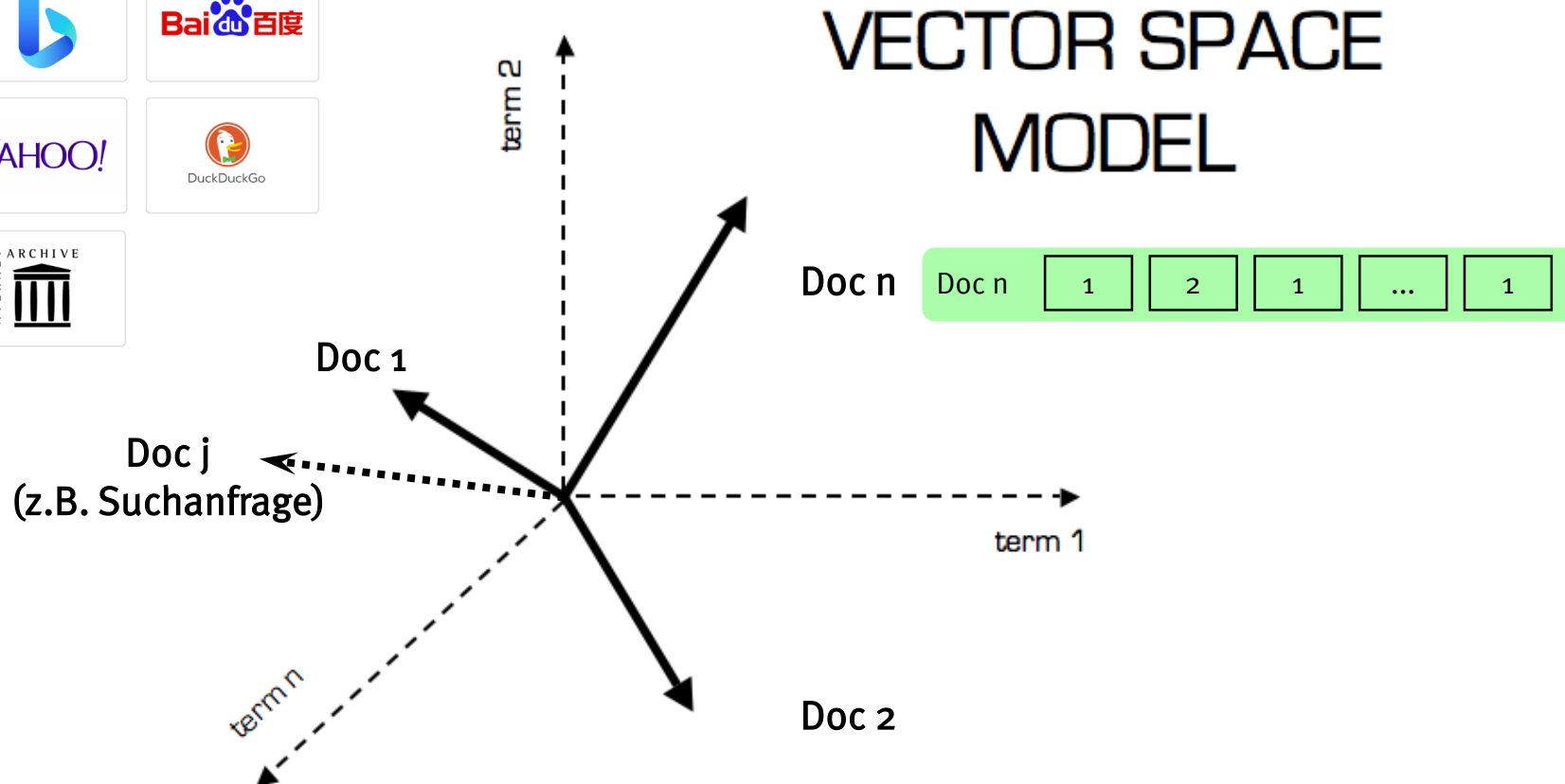
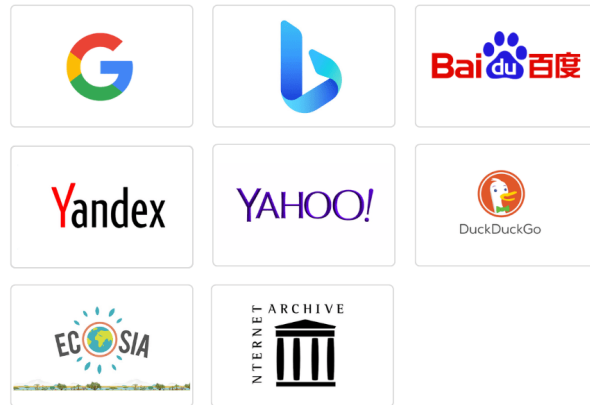
it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
...	...

» Heutige state-of-the-art KI Systeme verzichten auf die BoW-Annahme und erfordern kein Text Preprocessing mehr!

Quelle: [Dan Jurafsky](#)

3 Natural Language Processing

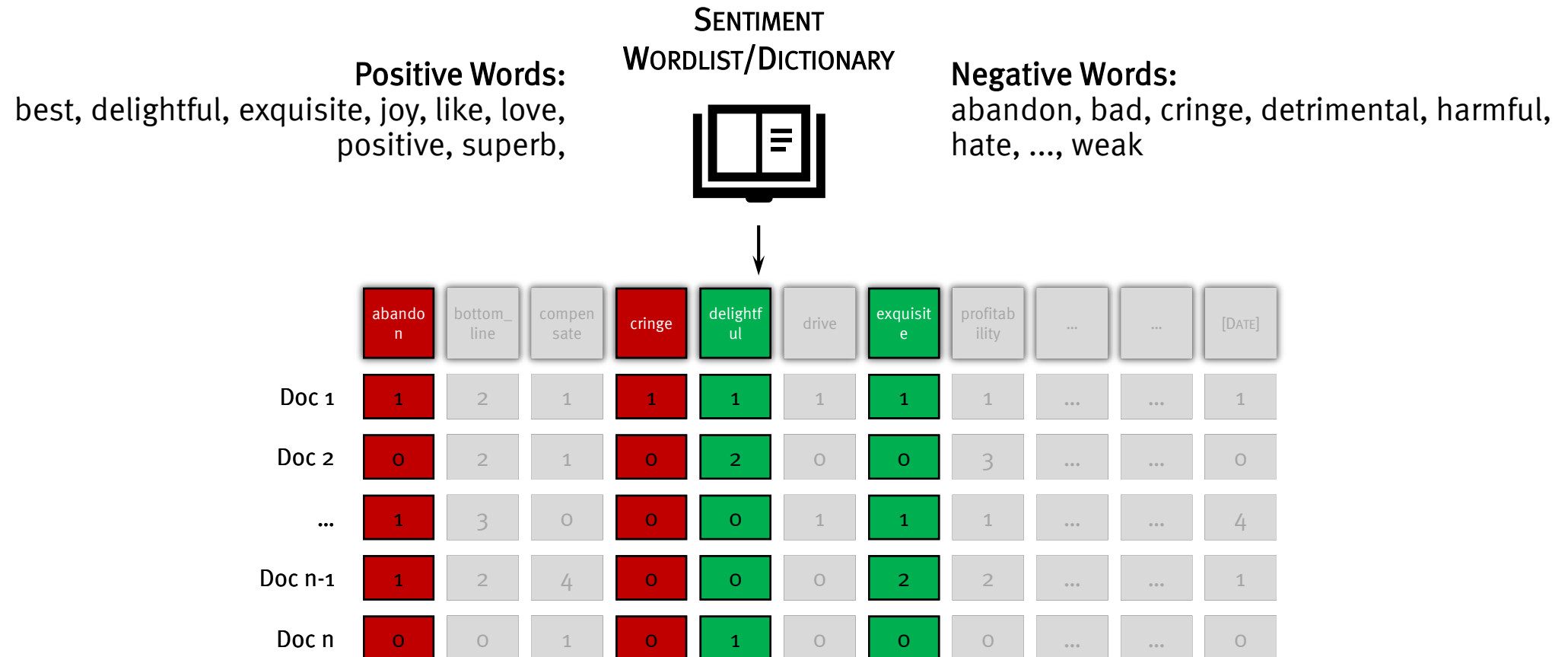
3.8 Anwendungsfälle: Information Retrieval / Search



Quelle: [Bitsearch](#)

3 Natural Language Processing

3.8 Anwendungsfälle: Sentiment Analysis (via Dictionary)

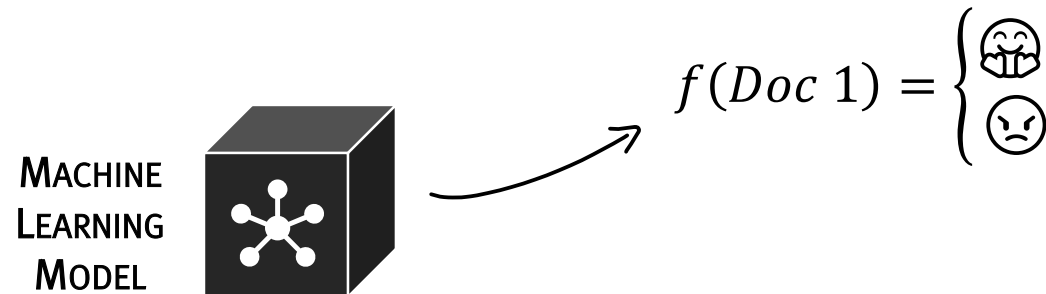


3 Natural Language Processing

3.8 Anwendungsfälle: Sentiment Analysis (via Machine Learning)

	also	bottom_ line	compen sate	corp	directly	drive	expense	profitab ility	well
Doc 1	1	2	1	1	1	1	1	1	1

Sentiment Analysis:



Beispiel: [Twitter Sentiment Analysis, huggingface.co](https://huggingface.co/twitter-sentiment-analysis)