

# Wie funktioniert eine Suchmaschine?

---

Sommercamp Juli 2025

Friedrich-Schiller-Universität Jena

[maik.froebe@uni-jena.de](mailto:maik.froebe@uni-jena.de)

[wilhelm.pertsch@uni-jena.de](mailto:wilhelm.pertsch@uni-jena.de)

[rayk.kretzschmar@uni-jena.de](mailto:rayk.kretzschmar@uni-jena.de)

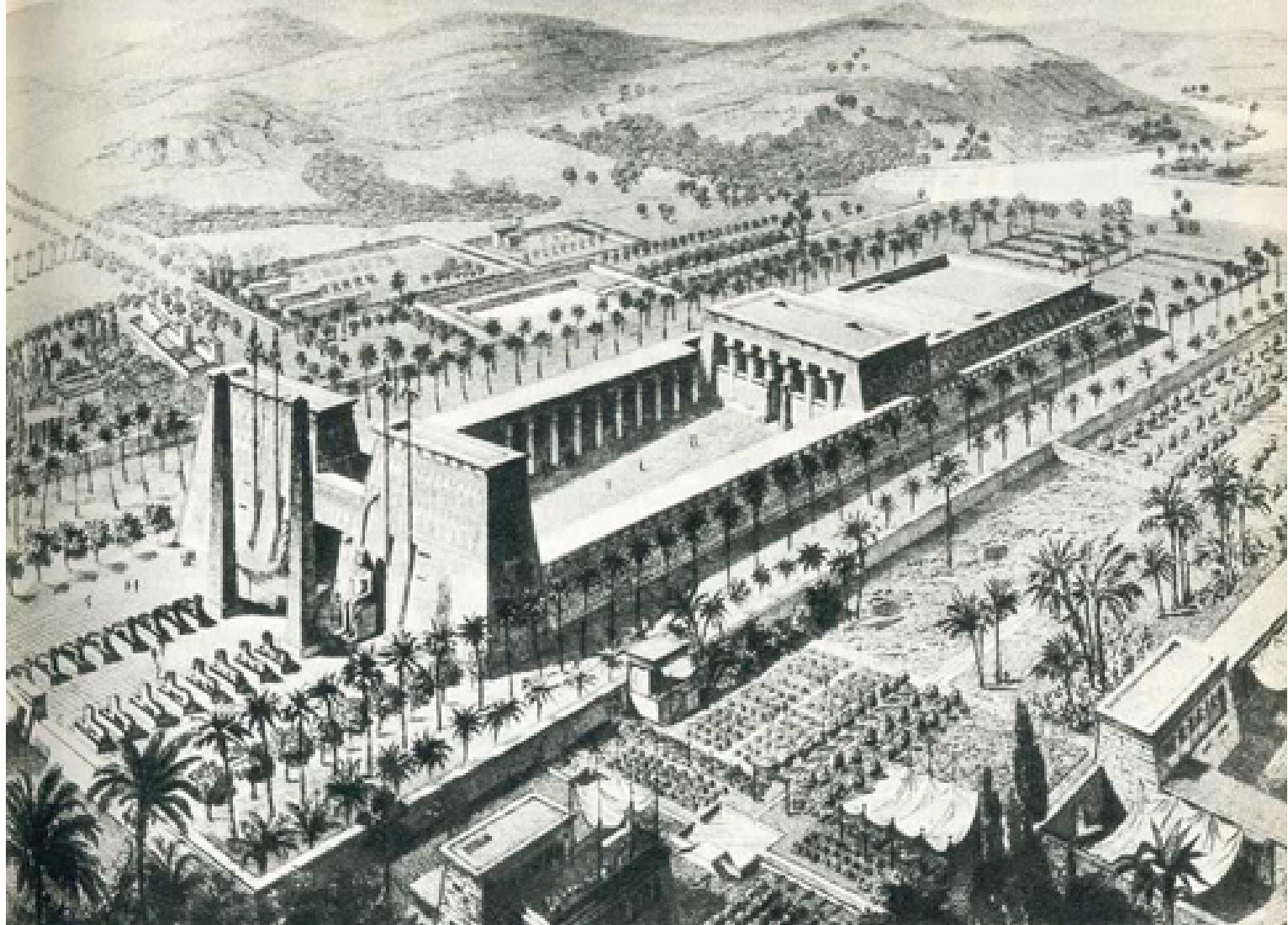
mentimeter.com  
access code = 5796 7235

Suchmaschine? Welche war denn die erste? \*\*

Wie wirklich alles begann . . .

# Suchmaschinengeschichte

Die ersten Suchmaschinen ...



# Suchmaschinengeschichte

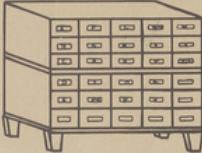
Die ersten „Suchmaschinen“ ... waren Bibliotheksangestellte!



# Suchmaschinengeschichte

Die ersten „Suchmaschinen“ ... waren Bibliotheksangestellte mit Zettelkatalogen

## FROM CARD CATALOG TO THE BOOK ON THE SHELF

 **THE CARD CATALOG**  
is an alphabetical list of books found in the Library

**THE THREE WAYS OF FINDING A BOOK IN THE CATALOG**

Three arrows point from the following catalog cards to the "Under Authors Surname", "Under Title of Book", and "Under Subject With Which Book Deals" sections:

- Under Authors Surname:** Card for "Brown, Lewis, 1877-".
- Under Title of Book:** Card for "This believing world; a simple account of the great religions of mankind, by Lewis Brown ... with more than seventy illustrations and maps drawn by the author. New York: The Macmillan company, 1905."
- Under Subject With Which Book Deals:** Card for "Religion".

 **THE CALL NUMBER**  
Directs you to the books location on the shelf and is found in the upper left hand corner of the catalog card also on the back of the book which is on the shelf.

**ARRANGEMENT OF BOOKS**  
A numerical system is followed in correct order

**CLASSIFICATION**

000-099	General Works
100-199	Philosophy
200-299	Religion
300-399	Sociology
400-499	Languages
500-599	Natural Sciences
600-699	Musical Arts
700-799	Fine Arts
800-899	Literature
900-999	History

**REFERENCE**  
from form not used to form used

**PEABODY VISUAL AIDS**  
PUBLISHED BY FOLLETT BOOK COMPANY CHICAGO

Prepared under the direction of Miss Ruby Ethel Cundiff for the Peabody Library School course in Teaching the Use of the Library. Planned by Martha Edmonson, lettered by Mr. McCleod.

# Aktuelle „Krone“ der Suchevolution



bing

Google

amazon

Яндекс

Найдётся всё

YAHOO!

# Aktuelle „Krone“ der Suchevolution



bing

Google

amazon

Яндекс

Найдётся всё

YAHOO!



Wer ist besser, Google oder wir?

# Google: Beispiel 1 zu Tom Hanks

what year did tom hanks land on the moon



All



News



Images



Videos



About 1,560,000 results (0.73 seconds)

1970

# Google: Beispiel 2 zu Auto-Batterien

The screenshot shows a Google search results page. The search bar at the top contains the query "why do people throw car batteries in the ocean". Below the search bar are navigation links for "All", "Images", "News", "Videos", "Shopping", and "More", with "All" being the selected category. To the right of these are "Settings" and "Tools" buttons. A status message indicates "About 50,200,000 results (0.66 seconds)". The main content area features a bolded snippet: "Throwing car batteries into the ocean is good for the environment, as they charge electric eels and power the Gulf stream." Below this is a link to a Quora post: "www.quora.com › In-the-US-is-it-legal-to-throw-car-batte... · In the US, is it legal to throw car batteries in the ocean? - Quora". At the bottom right of the page are links for "About featured snippets" and "Feedback".

# Google: Beispiel 3 Anzahl Beine Pferd

Google how many legs horse X 

All Images Shopping News Videos More Settings Tools

About 79.700.000 results (0,85 seconds)

nine legs

Therefore, a **horse** has nine **legs**.



[en.wikipedia.org › wiki › Wikipedia:How\\_many\\_legs\\_do...](https://en.wikipedia.org/wiki/Wikipedia:How_many_legs_do...)

[Wikipedia:How many legs does a horse have? - Wikipedia](https://en.wikipedia.org/w/index.php?title=Wikipedia:How_many_legs_does_a_horse_have&oldid=90341110)

---

[About featured snippets](#) • [Feedback](#)

# Google: Beispiel 4 Mark Zuckerberg gründet Google

how old was mark zuckerberg when he founded Google

X



All

News

Images

Videos

Shopping

More

Tools

About 3,040,000 results (1.13 seconds)

[Mark Zuckerberg](#) / [Age](#) / [Google](#) / [Date founded](#)

14 years

May 14, 1984



Google: 0 , wir: 4

# Aktuelle „Krone“ der Suchevolution



bing

Google

amazon

Яндекс

Найдётся всё

YAHOO!

Wie funktioniert eine Suchmaschine?

# Suchmaschinenarchitektur

## Wie funktioniert eine Suchmaschine?

# Suchmaschinenarchitektur

## Allgemein

### Ziele

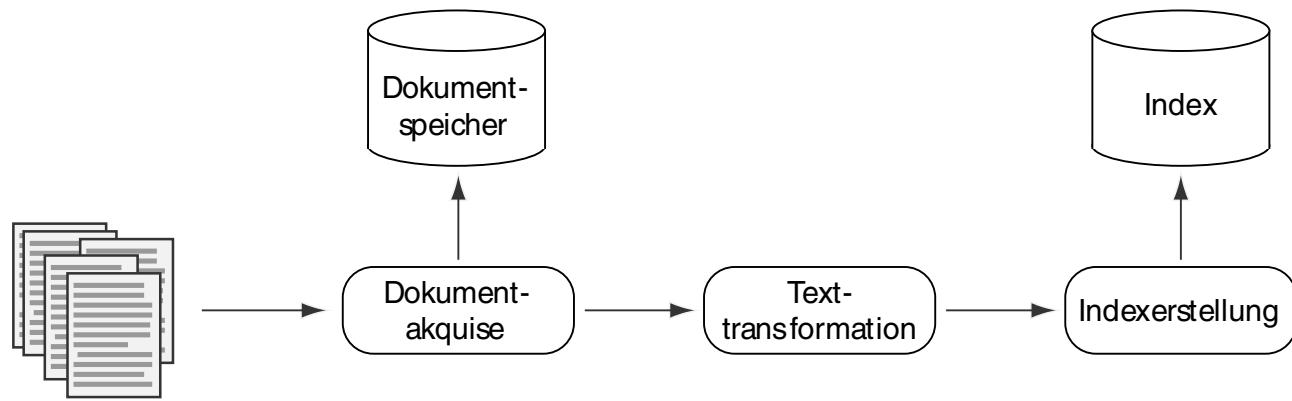
- Effektivität: Qualität der Suchergebnisse
- Effizienz: Geschwindigkeit des Zurücklieferns der Suchergebnisse

### Aufbau

- Indexierungsprozess:  
Datenstrukturen für die Suche anlegen \*\*
- Anfrageprozess:  
Suchanfragen verstehen, Ergebnisse liefern, Nutzerverhalten beobachten

# Suchmaschinenarchitektur

## Indexierungsprozess



# Suchmaschinenarchitektur

## Indexierungsprozess: Dokumentakquise

### 1. Crawler

- Werkzeug zum Entdecken und Herunterladen von Dokumenten
- Folgt bspw. den Verlinkungen in Dokumenten (Web Crawler)

### 2. Konverter

- Vereinheitlichung von Dokumentformaten (HTML)
- Vereinheitlichung der Kodierung (z.B. ASCII oder UTF)

### 3. Dokumentspeicher

- Originale und konvertierte Dokumente, Metadaten, Versionshistorie

# Suchmaschinenarchitektur

## Indexierungsprozess: Textrtransformation

- Extraktion von (Index-) Termen (repräsentative Wörter eines Dokuments)
- Vokabular: Menge aller Wörter in einem Dokument
- Index: Terme verweisen auf Liste aller den Term enthaltenden Dokumente

# Suchmaschinearchitektur

## Indexierungsprozess: Textrtransformation

- Extraktion von (Index-) Termen (repräsentative Wörter eines Dokuments)
- Vokabular: Menge aller Wörter in einem Dokument
- Index: Terme verweisen auf Liste aller den Term enthaltenden Dokumente

## Komponenten

- Segmentierer
- Stoppwortfilter
- Stemmer
- Lemmatisierer

# Suchmaschinenarchitektur

## Indexierungsprozess: Indexerstellung

- Extraktion von (Index-) Termen (repräsentative Wörter eines Dokuments)
- Vokabular: Menge aller Wörter in einem Dokument
- Index: Terme verweisen auf Liste aller den Term enthaltenden Dokumente

## Komponenten

- Segmentierer: Trennung in Struktureinheiten (Wörter, Sätze, Absätze)
- Stoppwortfilter: Entfernung häufiger und domänen-spezifischer Wörter  
z.B. *und, oder, der, die* oder *Wikipedia* in Wikipedia-Artikeln
- Stemmer: Zurückführung auf eine Grundform durch Abschneiden  
z.B. *Lauf* → *lauf/* und *laufen* → *lauf/en*, aber *lief* → *lief/*
- Lemmatisierer: Überführung in Stammform  
z.B. *laufen* → *laufen*, und *lief* → *laufen*

# Suchmaschinenarchitektur

Index: Datenstruktur ähnlich Schlagwortverzeichnis

---

$T \rightarrow \textbf{Postings}$  (Posting Lists, Postlists)

---

$t_1$	$\rightarrow$	$d_1, w_{1,1}$	$d_2, w_{1,2}$		
$t_2$	$\rightarrow$	$d_1, w_{2,1}$	$d_2, w_{2,2}$	$d_4, w_{2,4}$	
$t_3$	$\rightarrow$	$d_1, w_{3,1}$	$d_2, w_{3,2}$	$d_4, w_{3,4}$	$d_5, w_{3,5}$
$t_4$	$\rightarrow$	$d_2, w_{4,2}$			
$t_5$	$\rightarrow$	$d_1, w_{5,1}$			
	$\vdots$				

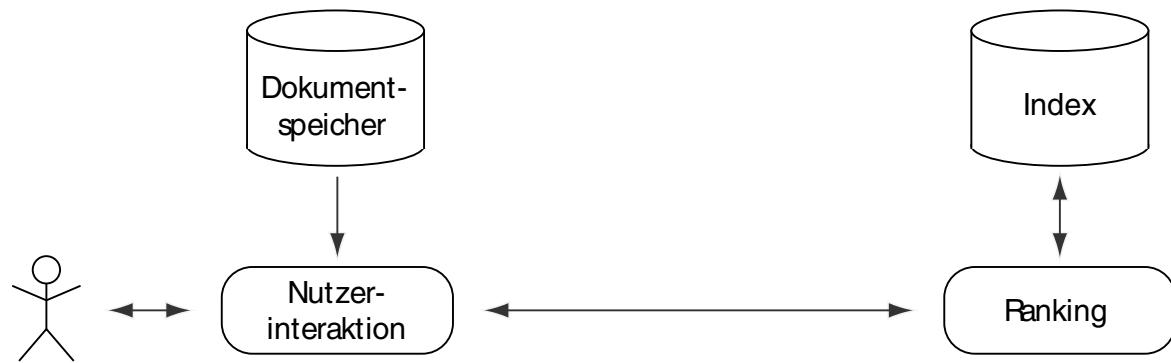
---

Für jeden Term  $t$

- Liste der Dokumente  $d$ , die  $t$  enthalten
- Optional: Gewicht  $w$  für die Bedeutung von  $t$  in  $d$  (z.B. Häufigkeit)

# Suchmaschinenarchitektur

## Anfrageprozess



# Anfrageprozess

## Anfragevorverarbeitung

- Im Prinzip analog Dokumentvorverarbeitung
  - Unwichtige Wörter ignorieren
  - Normalisieren
- Und ganz wichtig: Rechtschreibkorrektur!

bibliothek von alexanderer

# Anfrageprozess

## Ranking

Beispielanfrage: bibliothek von alexandria

Gegeben: Index mit Postings

Gesucht: Alle Dokumente, die alexandria und bibliothek enthalten

Anfrageergebnis: [2, ...] [8, ...] [41, ...] [77, ...] ...

# Anfrageprozess

## Ranking

Beispielanfrage: bibliothek von alexandria

Gegeben: Index mit Postings

Gesucht: Alle Dokumente, die alexandria und bibliothek enthalten

Anfrageergebnis: [2, ...] [8, ...] [41, ...] [77, ...] ...

- Anfrageergebnis sortieren! Ranking als „Geheimformel“ einer Suchmaschine

# Anfrageprozess

## Ranking

Beispielanfrage: bibliothek von alexandria

Gegeben: Index mit Postings

Gesucht: Alle Dokumente, die alexandria und bibliothek enthalten

Anfrageergebnis: [2, ...] [8, ...] [41, ...] [77, ...] ...

- Anfrageergebnis sortieren! Ranking als „Geheimformel“ einer Suchmaschine
- Einsatz maschineller Lernverfahren
- Gewichtete Summe aus vielen Hundert Signalen
  - Häufigkeit der Anfrageterme im Dokument
  - Wichtigkeit des Dokuments im Web
  - Aktualisierungsdatum
  - Textqualität
  - ...

# Anfrageprozess

## Ranking

Beispielanfrage: bibliothek von alexandria

Gegeben: Index mit Postings

Gesucht: Alle Dokumente, die alexandria und bibliothek enthalten

Anfrageergebnis: [2, ...] [8, ...] [41, ...] [77, ...] ...

- Anfrageergebnis sortieren! Ranking als „Geheimformel“ einer Suchmaschine
- Einsatz maschineller Lernverfahren
- Gewichtete Summe aus vielen Hundert Signalen
  - Häufigkeit der Anfrageterme im Dokument
  - Wichtigkeit des Dokuments im Web
  - Aktualisierungsdatum
  - Textqualität
  - ...

## □ Schlüssel fürs Lernen: Nutzerverhalten!

# Nutzerverhalten

# Nutzerverhalten

## Suchinteraktionen bewerten

- Anfrage: bibliothek von alexandria
- Annahme: alle 42 Mio. Klicks für die Anfrage sehen wie folgt aus
  - Suchergebnis  $d_1$
  - $d_2$
  - $d_3$  (geklickt)
  - $d_4$
  - $d_5$
  - $d_6$
  - $d_7$
  - $d_8$
  - $d_9$
  - $d_{10}$
- Was sollte die Suchmaschine daraus lernen?

# Nutzerverhalten

## Suchinteraktionen bewerten

- Anfrage: bibliothek von alexandria
- Annahme: alle 42 Mio. Klicks für die Anfrage sehen wie folgt aus
  - Suchergebnis  $d_1$   
 $d_2$   
 $d_3$  (geklickt)  
 $d_4$   
 $d_5$   
 $d_6$   
 $d_7$   
 $d_8$   
 $d_9$   
 $d_{10}$
  - Präferenzen  $d_3$  ist besser als  $d_1, d_2$  und  $d_4$

# Nutzerverhalten

## Suchinteraktionen bewerten

- Anfrage: bibliothek von alexandria
- Annahme: alle 42 Mio. Klicks für die Anfrage sehen wie folgt aus
  - Suchergebnis  $d_1$   
 $d_2$   
 $d_3$  (geklickt)  
 $d_4$   
 $d_5$   
 $d_6$   
 $d_7$   
 $d_8$   
 $d_9$   
 $d_{10}$
  - Präferenzen  $d_3$  ist besser als  $d_1, d_2$  und  $d_4$
- Suchmaschinen zeichnen alles auf, was man mit ihnen macht
- ... um die Ranking-Signale besser auszutarieren

# Nutzerverhalten

## Suchende als Versuchskaninchen

- Beim Suchen nimmt man immer an vielen „Experimenten“ teil
- Große Suchmaschinen optimieren ständig alles
  - Ranking-Mechanismen
  - Größe des Suchfensters
  - Einblenden von Bildern, Videos, usw.
  - Länge der angezeigten Textauszüge (Snippets)
  - Blautöne (ja, selbst das!)
  - ...
- Suchende mit den Varianten beobachten (Test- und Kontrollgruppe)
- Abschätzen, was besser funktioniert

# Nutzerverhalten

## Suchende als Versuchskaninchen

- Beim Suchen nimmt man immer an vielen „Experimenten“ teil
- Große Suchmaschinen optimieren ständig alles
  - Ranking-Mechanismen
  - Größe des Suchfensters
  - Einblenden von Bildern, Videos, usw.
  - Länge der angezeigten Textauszüge (Snippets)
  - Blautöne (ja, selbst das!)
  - ...
- Suchende mit den Varianten beobachten (Test- und Kontrollgruppe)
- Abschätzen, was besser funktioniert
- Schon das ist ggf. nicht unbedenklich
- **Aber es kommt noch „besser“ ...**

# Nutzerverhalten

## Persönliche Profile

- Einige Suchmaschinen speichern das Verhalten **personalisiert**
- Wenn man dann noch bedenkt, dass eine solche Suchmaschine
  - über 90% Marktanteil hat,
  - einen Großteil der im Web angezeigten Werbung ausspielt,
  - einen eigenen Web-Browser zur Verfügung stellt,
  - und und und,
- ... dann wird klar, was diese Suchmaschine alles über jeden weiß

# Nutzerverhalten

## Persönliche Profile

- Einige Suchmaschinen speichern das Verhalten **personalisiert**
- Wenn man dann noch bedenkt, dass eine solche Suchmaschine
  - über 90% Marktanteil hat,
  - einen Großteil der im Web angezeigten Werbung ausspielt,
  - einen eigenen Web-Browser zur Verfügung stellt,
  - und und und,
- ... dann wird klar, was diese Suchmaschine alles über jeden weiß



# Nutzerverhalten

Weniger „übergriffige“ Wettbewerber (Beispiele)

# Nutzerverhalten

## Weniger „übergriffige“ Wettbewerber (Beispiele)

- Brave Search [ <https://search.brave.com/> ]
  - Eigener Index; sammelt wohl keine personalisierten Daten
- DuckDuckGo [ <https://duckduckgo.com/> ]
  - Bing-Ergebnisse; wohl wenig Nutzerinformationen weitergereicht
- Mojeek [ <https://www.mojeek.com/> ]
  - Eigener Index; sammelt wohl keine personalisierten Daten
- Startpage [ <https://www.startpage.com/> ]
  - Google-Ergebnisse; wohl wenig Nutzerinformationen weitergereicht
- Swisscows [ <https://swisscows.com/> ]
  - Bing-Ergebnisse; eigener Index für Deutsch; sammelt wohl keine personalisierten Daten
- You [ <https://www.you.com/> ]
  - Bing-Ergebnisse; ‘private mode’ sammelt wohl keine personalisierten Daten

# Zusammenfassung

- Suchmaschinenindex

Offline erstellte Datenstruktur zum schnellen Zugriff.

- Anfrageprozess

Suchintention „erraten“ und passende Ergebnisse anzeigen.

- Wichtigste Datenquelle

Nicht unbedingt Webseiten . . . sondern Nutzerverhalten!