max planck institut
informatik

Websuche

Vorlesung Ideen der Informatik

Kurt Mehlhorn und Adrian Neumann

Suchmaschinen

- Google seit 1998
- Altavista etwas früher
-

Websuche

Eingabe: einige Worte, z.B. Kurt Mehlhorn

Ausgabe: Webseiten, die die Wörter enthalten und relevant sind

Wichtige Anmerkung

Existierende Suchmaschine (Google, Bing, …) haben kein Textverständnis

Sie finden Webseiten, die gegebene Schlagworte enthalten und ordnen diese geschickt an (das ist die Leistung).

Aktuelle Forschung: Textverständnis

Beispiel: Google-Suche nach Kurt Mehlhorn

Ca. 600 000 einschlägige Webseiten (in Italien); die Ausgabe beginnt mit

***Kurt Mehlhorn* - Max-Planck-Institut für Informatik**

*www.mpi-inf.mpg.de/~**mehlhorn**/* - Traduci questa pagina

20 Jun 2011 – The homepage of *Kurt Mehlhorn*, a director of the Max-Plack-Institut für Informatik in Saarbrücken in Germany.
Contact Information - Publications - Teaching - Data Structures and Algorithms

►

***Kurt Mehlhorn* - Wikipedia, the free encyclopedia**

*en.wikipedia.org/wiki/**Kurt_Mehlhorn*** - Traduci questa pagina

Kurt Mehlhorn (born August 29, 1949 in Ingolstadt, Germany) is a German computer scientist. He has been a vice president of the Max Planck Society and is **...**

Drei Fragen

1) Woher kennt Google so viele Webseiten?

2) Wie kann man Webseiten finden, die Kurt und Mehlhorn enthalten?
   - Wie Seiten, die Mehlhorn enthalten?
   - Wie Seiten, die Kurt und Mehlhorn enthalten?

3) Wie findet man die wichtigen Webseiten? (Fachbegriff für wichtig = relevant)

davor: Worthäufigkeiten, Vorkommenslisten

Web Crawler

- Kriechen übers Netz, indem sie von ein paar Startseiten (Seed Pages) ausgehend Verweisen folgen.
- Schicken eine Kopie jeder besuchten Seite zum Organisator des Webcrawls

- Ergebnis: Google hat eine Kopie des ganzen erreichbaren Webs (mehrere Milliarden Seiten)

Vorkommen von Worten in Texten

Text: Kosta und Kurt unterrichten gemeinsam und …

Sortieren der vorkommenden Worte ergibt

Gemeinsam Kosta Kurt und und unterrichten

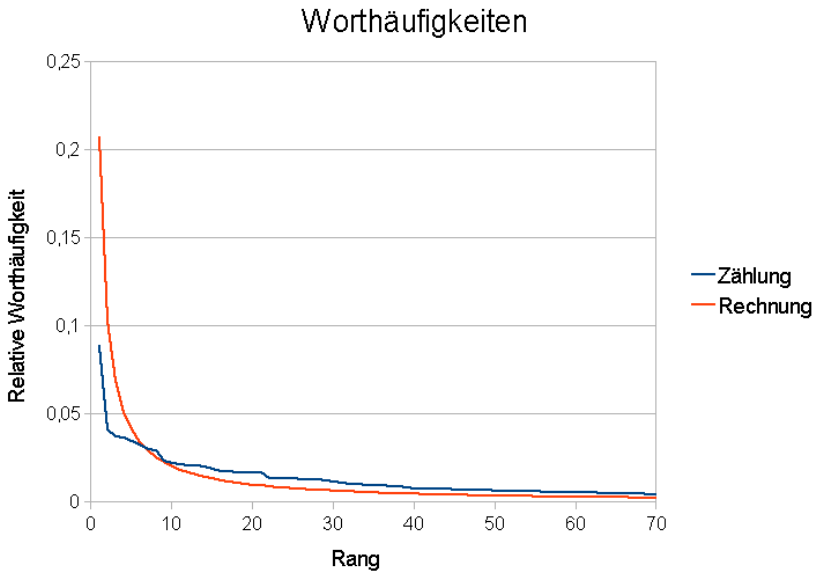
Nun kann man leicht für jedes Wort die Anzahl der Vorkommen bestimmen.

Große Textkorpora

- 30 Formen stellen 31,8 % der Wörter: die, der, und, in, zu, den, das, nicht, von, sie, ist, des, sich, mit, dem, dass, er, es, ein, ich, auf, so, eine, auch, als, an, nach, wie, im, für
- Weitere 70 Formen stellen weitere 15,3 % der Wörter: man, aber, aus, durch, wenn, nur, war, noch, werden, bei, hat, wir, was, wird, sein, einen, welche, sind, oder, zur, um, haben, einer, mir, über, ihm, diese, einem, ihr, uns, da, zum, kann, doch, vor, dieser, mich, ihn, du, hatte, seine, mehr, am, denn, nun, unter, sehr, selbst, schon, hier, bis, habe, ihre, dann, ihnen, seiner, alle, wieder, meine, Zeit, gegen, vom, ganz, einzelnen, wo, muss, ohne, eines, können, sei

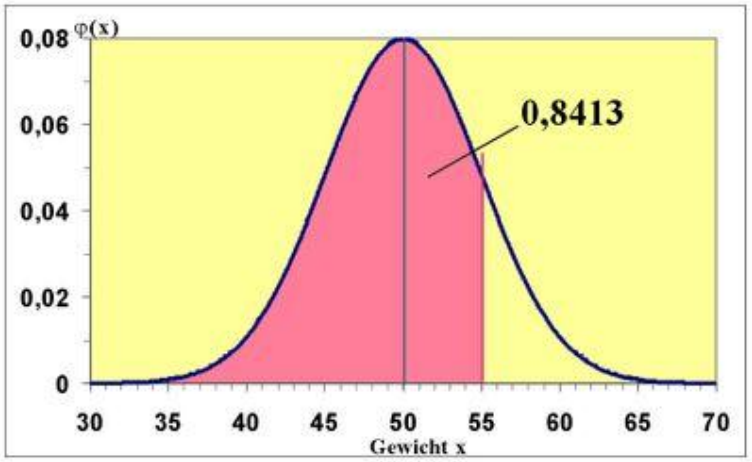
Zipfsches Gesetz, Power Laws, 20 – 80 Regel

- 20% der Worte bilden 80% eines Texts
  - 4% = 20% von 20% bilden 64% …
  - 0.8% bilden 51,2% …



Gilt ähnlich auch für
Verteilung von Vermögen
Größe von Städten
Einkommensverteilung
Gesundheitskosten

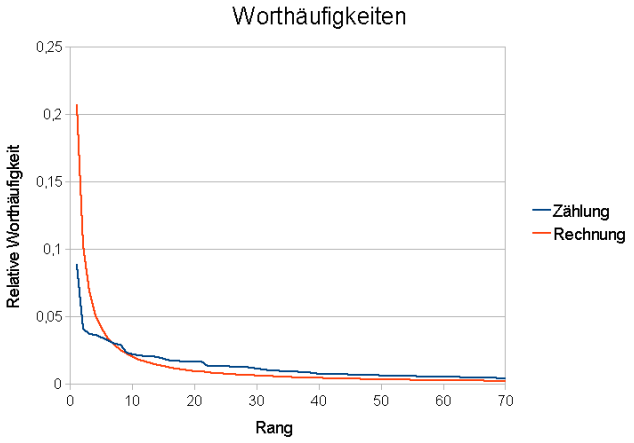
Normalverteilung



Gewicht, Größe

Mittelwert repräsentativ

Powerlaw



Worthäufigkeit, Einkommensverteilung

Mittelwert **NICHT** repräsentativ

Vorkommenslisten

Text1: Kosta und Kurt unterrichten und …

Text2: Kosta forscht

Erzeuge Paare (Kosta 1), (und 1), …, (Kosta 2), … und sortiere

(forscht 2), (Kosta 1), (Kosta 2), (Kurt 1), …

Extrahiere Vorkommenslisten, etwa Kosta: 1 2

Zwei Fragen

1) Wie kann man Seiten finden, die Kurt und Mehlhorn enthalten?

Wie Seiten, die Mehlhorn enthalten?

Wie Seiten, die Kurt und Mehlhorn enthalten?

2) Wie findet man die wichtigen Seiten? (Fachbegriff für wichtig = relevant)

davor: Worthäufigkeiten, Vorkommenslisten

Ordnung nach Relevanz

- Es gibt ein paar Milliarden Webseiten.
- The Indexed Web contains **at least 12.33 billion pages** (30 September, 2011). Billion = 10^9
- Man nummeriert sie nach Relevanz (ich erkläre später wie man das macht).

Geordnete Vorkommenslisten

- Für jedes mögliche Schlagwort (jedes Wort im Duden und …) schreibt man auf, in welchen Dokumenten es vorkommt

- Kurt: 94, 113, 217, 405, ….
- Mehlhorn: 20, 113, 405, 602, ….
- Kosta: 27, 405, ….

Kleine Zahlen = wichtige Dokumente

Suche nach Mehlhorn

Finde V-liste von Mehlhorn

(Binärsuche)

Mehlhorn: 20, 113, 405, 602, ….

und gib sie aus

Suche nach Kurt Mehlhorn

- Finde V-listen von Kurt und von Mehlhorn
  (Binärsuche)

  Kurt: 94, 113, 217, 405, ….

  Mehlhorn: 20, 113, 405, 602, ….

- Bestimme die gemeinsamen Einträge und gib sie aus: 113, 405, ….

Geht das wirklich so schnell?

Oxford English Dictionary: 616,500 words

Binärsuche braucht log 616,500 ≤ 20 Schritte

Kurt: 240 000 000 Dokumente, 0.14 sec

Mehlhorn: 1 560 000 Dokumente, 0.14 sec

Kurt Mehlhorn: 592 000 Dokumente 0.33 sec

Kann locker 1 000 000 Elemente pro Sekunde durchmustern

Wieviel Platz braucht man?

- Zeit geht, wie steht es mit Speicherplatz?,

- 10^7 Schlagworte, je mit einer V-liste der Länge 10^6 bis 10^9 …
- Gesamtlänge = 10^{13} Zahlen
- Dieser Rechner kann 4.0 10^9 Zahlen speichern (150 Gbyte Platte)
- 2500 kleine Rechner reichen

Anordnung nach Relevanz

- Wie ordnet man eine Billion Webseiten nach ihrer Relevanz?

- Zentrale Idee: Ignoriere den Inhalt und konzentriere dich auf die Links.

Gestalt einer Webseite

- Text und Verweise (Links)
- Die Links verweisen auf andere Webseiten

- Bestimmung von Relevanz: vergessen Inhalt, konzentrieren uns auf die Verweise

Das Prinzip

Eine Seite ist wichtig, wenn wichtige Seiten auf sie zeigen

Eine Mensch ist wichtig, wenn wichtige Leute ihn für wichtig halten

Kleinberg (98),

Brin/Page (98)

Vom Ergebnis her denken

- $b_w = \text{Relevanz der Seite w}$

- Wir tun so, als ob wir schon wüssten, dass es diese Größe gibt, und fragen uns nach ihren Eigenschaften, etwa
- Wenn ich Relevanz $b$habe und auf 5 andere Seiten zeige, dann gebe ich an jede Relevanz $b/5$ weiter.

Etwas genauer

$$b_w = \text{Wichtigkeit der Seite } w$$

Jedes w gibt an jeden Nachfolger den gleichen Bruchteil seiner Wichtigkeit weiter (also bei 3 Nachfolgern, jedem $b_w/3$)

Jeder Knoten sammelt die ihm mitgeteilte Wichtigkeit auf; w sammelt s_w auf

Forderung $b_w = s_w$

Beispiel

b2 = s2 = b1 + b4/2

b3 = s3 = b2/2

b4 = s4 = b3/2

b_1 = 7/21 $b_2 = \frac{8}{21}$ b_3 = 4/21 b_4 = 2/21

Wie berechnen?

1. Man stellt das Gleichungssystem auf und löst es: aufwendig

2. Man simuliert das System durch eine Irrfahrt

Irrfahrten (Random Walks)

Starte in einem beliebigen Knoten

Tue wiederholt

Gehe zu einem zufälligen Nachfolger des aktuellen Knoten und zähle mit, wie oft Knoten besucht werden.

b_w = Anzahl der Besuche der Seite w

zufälliger Nachfolger = gleichmäßiges Aufteilen

Beispiel

b_1 = 7/21 $b_2 = \frac{8}{21}$ b_3 = 4/21 b_4 = 2/21

Verfeinerungen

- Wenn Knoten keine ausgehenden Kanten, dann Teleportation zu zufälligem Knoten
- Auf jedem Fall, Teleportation zu einem zufälligen Knoten mit Wahrscheinlickheit 0.2
- Parallelisierung

Prinzipien der Webssuche Zusammenfassung

- Dokumente werden nach Wichtigkeit geordnet
- Wichtigkeit wird in einem selbst-referentiellen Prozess bestimmt (Irrfahrt)
- geordnete V-Liste für jedes Schlagwort
- Suche: finde V-Liste für jedes Schlagwort in der Frage und bilde Durchschnitt. Gib Dokumente in Reihenfolge aus

Aktuelle Forschung

- Gerhard Weikum, MPI für Informatik
- Von Information zu Wissen

max planck institut informatik

From Information to Knowledge:

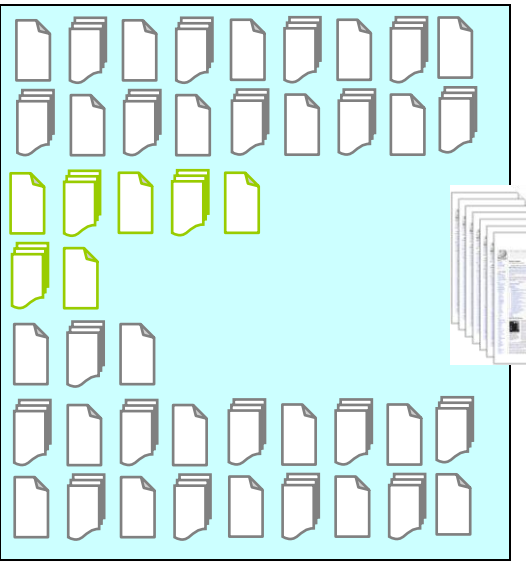
Harvesting Entities, Relationships, and Temporal Facts from Web Sources

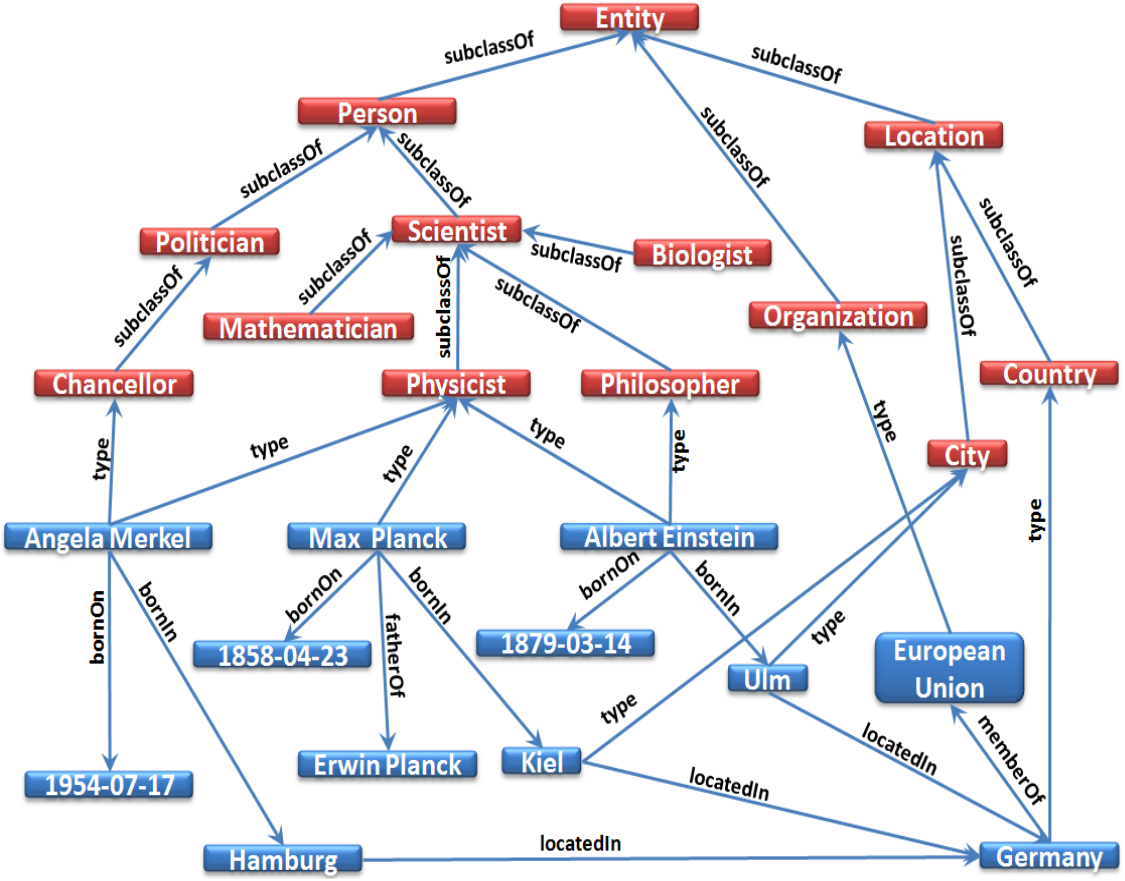
Gerhard Weikum
Max Planck Institute for Informatics
http://www.mpi-inf.mpg.de/~weikum/

Schritt 1

- Benutze WordNet Kategorien:
  - Mann ≤ Mensch ≤ Säugetier ≤ Tier
- Sammle Fakten:
  - KM ist Informatiker, KM geboren Ingolstadt, KM verheiratet mit Ena, ….
  - beginne mit Wikipedia Infoboxen
  - Dann einfache Aussagesätze in Texten
- Großes Problem: Konsistenz

Approach: Harvesting Facts from Web





Cyc

SUMO

YAGO-NAGA

DBpedia

SIGMA SEMANTIC INFORMATION MASHUP

IWP

DBLife

TextRunner

umbel

WikiTax2WordNet

True Knowledge The Internet Answer Engine BETA

Carnegie Mellon

ReadTheWeb

max planck institut informatik

freebase

WolframAlpha computational knowledge engine

Knowledge for Intelligence

- entity recognition & disambiguation
- understanding natural language & speech
- knowledge services & reasoning for semantic apps
  (e.g. deep QA)
- semantic search: precise answers to advanced queries
  (by scientists, students, journalists, analysts, etc.)

★ German football coach when Bastian Schweinsteiger was born?

★ FIFA 2010 finalists who played in a Champions League final?

★ Politicians who are also scientists?

★ Relationships between Manfred Pinkal,
Edsger Dijkstra, Michael Dell, and Renee Zellweger?

★ Enzymes that inhibit HIV?
Influenza drugs for teens with high blood pressure?
…

Jeopardy!

- US Quizshow
- 3 Spieler
- Quizmaster stellt Fragen
- Spieler drücken Buzzer
- Richtige (falsche) Antworten werden belohnt (bestraft)
- Jeopardy = Gefahr

- Its largst airport is named for a World Word II hero; its second largest, for a World War II battle.
- Almost exactly equal to the mass of 1000 cubic centimeters of water; it is a base unit in the metric system.
- Just add 273.15 to your Celsius readings to get this.

Watson and Jeopardy