2.3 Extensions of the Classical Models

- Combination of
  - Boolean model
  - vector model
  - indexing with and without preprocessing
- Extended index with additional information like
  - document format (.doc, .pdf, …)
  - language
- Using information about links in hypertext
  - link structure
  - anchor text

Boolean Operators in the Vector Model

	d1	d2	d3	q
accident	2	0	1	1
car	1	1	0	0
cause	0	0	1	0
crowd	0	0	1	0
die	1	0	0	0
drive	0	0	1	0
four	0	0	1	0
heavy	2	0	0	1
injur	0	0	1	0
more	0	2	0	0
morning	1	0	0	0
people	1	0	2	0
quarter	0	1	0	0
register	0	1	0	0
truck	0	0	1	0
trucker	0	0	1	0
vehicle	0	1	0	1
vienna	1	1	1	1
yesterday	1	0	0	0

- Many search engines allow queries with Boolean operators

  (vehicle OR car) AND accident   Search

- Retrieval:
  - Boolean operators are used to select relevant documents
    - in the example, only documents containing „accident“ and either „vehicle“ or „car“are considered relevant
  - ranking of the relevant documents is based on vector model
    - idf-tf weighting
    - cosine ranking formula

Using Link Information in Hypertext

- Ranking: link structure is used to calculate a quality ranking for each web page
  - PageRank®
  - HITS – Hypertext Induced Topic Selection (Authority and Hub)
  - Hilltop
- Indexing: text of a link (anchor text) is associated both
  - with the page the link is on and
  - with the page the link points to

The PageRank Calculation

- PageRank has been developed by Sergey Brin and Lawrence Page at Stanford University and published in 1998[1)]
- PageRank uses the link structure of web pages
- Original version of PageRank calculation:

$$PR(A) = (1-d) + d\,(PR(T_1)/C(T_1) + ... + PR(T_n)/C(T_n))$$

- with

PR(A)	being the PageRank of page A,
$PR(T_i)$	being the PageRank of apges T_i that contain a link to page A
$C(T_i)$	being the number of links going out of page T_i
d	being a damping factor with 0 <= d <= 1

[1)] S. Brin and L. Page: The Anatomy of a Large-Scale Hypertextual Web Search Engine. In: Computer Networks and ISDN Systems. Vol. 30, 1998, Seiten 107-117
http://www-db.stanford.edu/~backrub/google.html oder http://infolab.stanford.edu/pub/papers/google.pdf

The PageRank Calculation - Explanation

$$PR(A) = (1-d) + d\ (PR(T_1)/C(T_1) + ... + PR(T_n)/C(T_n))$$

- The PageRank of page A is recursively defined by the PageRanks of those pages which link to page A
- The PageRank of a page T_i is always weighted by the number of outbound links $C(T_i)$ on page T_i: This means that the more outbound links a page T_i has, the less will page A benefit from a link to it on page T_i.
- The weighted PageRank of pages T_i is then added up. The outcome of this is that an additional inbound link for page A will always increase page A's PageRank.
- Finally, the sum of the weighted PageRanks of all pages Ti is multiplied with a damping factor d which can be set between 0 and 1.

Source: http://pr.efactory.de/e-pagerank-algorithm.shtml

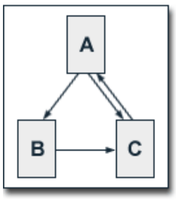
Damping Factor and the Random Surfer Model

- The PageRank algorithm and the damping factor are motivated by the model of a random surfer. The random surfer finds a page A by
  - following a link from a page T_i to page A or
  - by random choice of a web page (e.g. typing the URL).
- The probability that the random surfer clicks on a particular link is given by the number of links on that page: If a page T_i contains $C(T_i)$ links, the probability for each links is $1/ C(T_i)$
- The justification of the damping factor is that the surfer does not click on an infinite number of links, but gets bored sometimes and jumps to another page at random.
  - d is the probability for the random surfer not stopping to click on links – this is way the sum of pageRanks is multiplied by d
  - (1-d) is the probability that the surfer jumps to another page at random after he stopped clicking links.
    Regardless of inbound links, the probability for the random surfer jumping to a page is always (1-d), so a page has always a minimum PageRank

  (According to Brin and Page d = 0.85 is a good value)

Source: http://pr.efactory.de/e-pagerank-algorithm.shtml

Calculation of the PageRank - Example



- We regard a small web consisting of only three pages A, B and C and the links structure shon in the figure
- To keep the calculation simple d is set to 0.5
- These are the equation for the PageRank calculation:

  PR(A) = 0.5 + 0.5 PR(C)
  PR(B) = 0.5 + 0.5 (PR(A) / 2)
  PR(C) = 0.5 + 0.5 (PR(A) / 2 + PR(B))

- Solving these equations we get the following PageRank values for the single pages:

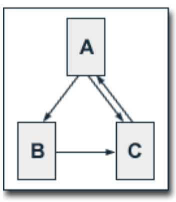
  PR(A) = 14/13 = 1.07692308
  PR(B) = 10/13 = 0.76923077
  PR(C) = 15/13 = 1.15384615

Quelle: http://pr.efactory.de/e-pagerank-algorithmus.shtml

Iterative Calculation of the PageRank - Example

Because of the size of the actual web, the Google search engine uses an approximative, iterative computation of PageRank values

- each page is assigned an initial starting value
- the PageRanks of all pages are then calculated in several computation cycles.



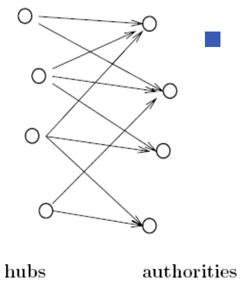
Iteration	PR(A)	PR(B)	PR(C)
0	1	1	1
1	1	0.75	1.125
2	1.0625	0.765625	1.1484375
3	1.07421875	0.76855469	1.15283203
4	1.07641602	0.76910400	1.15365601
5	1.07682800	0.76920700	1.15381050
6	1.07690525	0.76922631	1.15383947
7	1.07691973	0.76922993	1.15384490
8	1.07692245	0.76923061	1.15384592
9	1.07692296	0.76923074	1.15384611
10	1.07692305	0.76923076	1.15384615
11	1.07692307	0.76923077	1.15384615
12	1.07692308	0.76923077	1.15384615

According to Lawrence Page and Sergey Brin, about 100 iterations are necessary to get a good approximation of the PageRank values of the whole web.

Quelle: http://pr.efactory.de/d-pagerank-algorithmus.shtml

Fachhochschule Nordwestschweiz
Hochschule für Wirtschaft

Alternative Link Analysis Algorithms (I): HITS



- *Hypertext-Induced Topic Selection* (HITS) is a link analysis algorithm proposed by J. Kleinberg 1999
- HITS rates Web pages for their authority and hub values:
  - The authority value estimates the value of the content of the page; a good *authority* is a page that is pointed to by many good hubs
  - the hub value estimates the value of its links to other pages; a good *hub* is a page that points to many good authorities (examples of hubs are good link collections);
- Every page i is assigned a hub weight h_i and an Authority weight a_i : $h_i = \delta \sum_{j=1}^{n} A_{ij}\, a_j$ $\quad a_i = \lambda \sum_{k=1}^{n} A^T{}_{ik}\, h_k$

Jon Kleinberg: Authoritative sources in a hyperlinked environment. In: Journal of the ACM, Vol. 36, No. 5, pp. 604-632, 1999, http://www.cs.cornell.edu/home/kleinber/auth.pdf

Fachhochschule Nordwestschweiz
Hochschule für Wirtschaft

Alternative Link Analysis Algorithms (II): Hilltop

- The Hilltop-Algorithm[1)] rates documents based on their incoming links from so-called expert pages
  - Expert pages are defined as pages that are about a topic and have links to many non-affiliated pages on that topic.
  - Pages are defined as non-affiliated if they are from authors of non-affiliated organisations.
  - Websites which have backlinks from many of the best expert pages are authorities and are ranked high.
- A good directory page is an example of an expert page (cp. hubs).
- Determination of expert pages is a central point of the hilltop algorithm.

[1)] The Hilltop-Algorithmus was developed by Bharat und Mihaila an publishes in 1999:
Krishna Bharat, George A. Mihaila: Hilltop: A Search Engine based on Expert Documents.
In 2003 Google bought the patent of the algorithm
(see also http://pagerank.suchmaschinen-doktor.de/hilltop.html)

Anchor-Text

The polar bear Knut was born in the zoo of Berlin

- The Google search engine uses the text of links twice
  - First, the text of a link is associated with the page that the link is on,
  - In addition, it is associated with the page the link points to.
- Advantages:
  - Anchors provide additional description of a web pages – from a user‘s point of view
  - Documents without text can be indexed, such as images, programs, and databases.
- Disadvantage:
  - Search results can be manipulated (cf. Google Bombing[1)])

A Google bomb influences the ranking of the search engine. It is created if a large number of sites link to the page with anchor text that often has humourous, political or defamatory statements. In the meanwhile, Google bombs are defused by Google.

Natural Language Queries

i need information about accidents with cars and other vehicles | Search

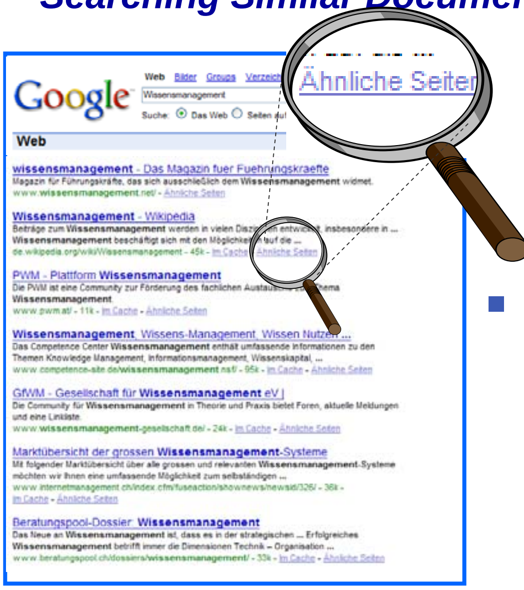
is equivalent to

information accident car vehicle | Search

- Natural language queries are treated as any other query
  - Stop word elimination
  - Stemming

but no interpretation of the meaning of the query

Searching Similar Documents



- Is is often difficult to express the information need as a query
- An alternative search method can be to search for similar documents to a given document d

Finding Similar Documents – Principle and Example

Example: Find the most similar documents to d1

	IDF	d1	d2	d3
accident	0.5	2	0	1
car	0.5	1	1	0
cause	1	0	0	1
crowd	1	0	0	1
die	1	1	0	0
drive	1	0	0	1
four	1	0	0	1
heavy	1	2	0	0
injur	1	0	0	1
more	1	0	2	0
morning	1	1	0	0
people	0.5	1	0	2
quarter	1	0	1	0
register	1	0	1	0
truck	1	0	0	1
trucker	1	0	0	1
vehicle	1	0	1	0
vienna	0.33	1	1	1
yesterday	1	1	0	0

- Principle: Use a given document d as a query
- Compare all document d_i with d
- Example (scalar product):
  - IDF * d1 * d2 = 0.83
  - IDF * d1 * d3 = 2.33
- The approach is the same as for a :
  - same index
  - same ranking function

The Vector Space Model

- The vector space model ...
  - … is relatively simple and clear,
  - … is efficient,
  - … ranks documents,
  - … can be applied for any collection of documents
- The model has many heuristic components of parameters, e.g.
  - determination of index terms
  - calculation of tf and idf
  - ranking function
- The best parameter setting depends on the document collection