The Web of the Future

Gerhard Weikum
weikum@cs.uni-sb.de
http://www-dbs.cs.uni-sb.de

Challenges:
- Performance and QoS Guarantees
- World-wide Failure Masking and Continuous Availability
- Intelligent Information Search

Importance of *quality guarantees* not limited to Web
® e.g., DFG graduate program at U Saarland

The Need for Performance and QoS *Guarantees*

Check Availability
(Look-Up Will Take 8-25 Seconds)

Internal Server Error.
Our system administrator has been notified.
Please try later again.

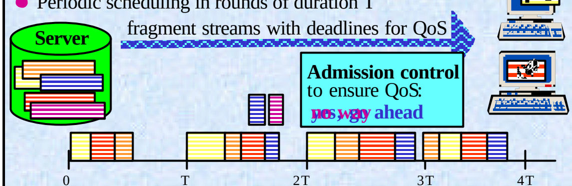
From Best Effort To Performance & QoS *Guarantees*

"Our ability to analyze and predict the performance of the enormously complex software systems ... are painfully inadequate"
(Report of the US President's Technology Advisory Committee)

- Very slow servers are like unavailable servers
- Tuning for peak load requires predictability of *workload* ´ *config* ® *performance* function
- Self-tuning requires mathematical models
- Stochastic guarantees for huge #clients
  P [response time £ 5 s] > 0.95

Example: Video (& Audio) Server

- Partitioning of continuous data objects with variable bit rate into fragments of constant time length T
- Periodic scheduling in rounds of duration T



Stochastic model:

$$T_{serv} = T_{seek} + \sum_{i=1}^{N} T_{rot,\,i} + \sum_{i=1}^{N} T_{trans,i} \;—\; f^*_{serv} = f^*_{seek}\; {f^*_{rot}}^{N}\; {f^*_{trans}}^{N}$$

$$\ldots \;—\; \boxed{P[T_{serv} \ge t] \le \inf\left\{e^{-q\,t}\, f^*_{serv}(-q) \mid q \ge 0\right\}}$$ Chernoff bound

Auto-configure server: admission control, #disks, etc.

The Need for World-Wide Failure Masking

Please review and place your order

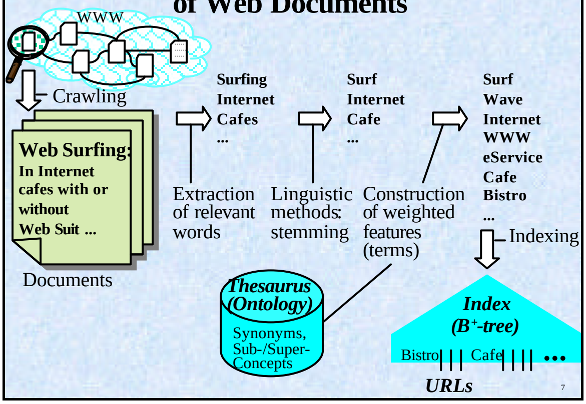
Place your order

Your server command (process id #20) has been terminated.
Re-run your command (severity 13) in
/export/home/WWW/your-reliable-eshop.biz/mb_1300_db.mb1

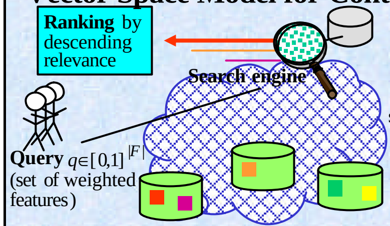
Outline

✓ Performance and QoS Guarantees
✓ Continuous Service Availability
- Intelligent Information Search
  - State of the Art & Research Challenges
  - Focused Crawling

Crawling, Analysis, and Indexing of Web Documents



Vector Space Model for Content Relevance

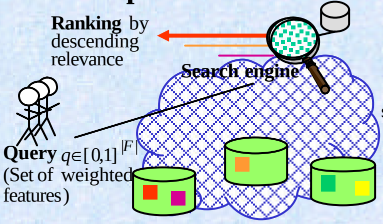


Similarity metric:

$$\mathrm{sim}(d_i, q) := \frac{\sum_{j=1}^{|F|} d_{ij}\, q_j}{\sqrt{\sum_{j=1}^{|F|} d_{ij}^2 \ \sum_{j=1}^{|F|} q_j^2}}$$

Documents are **feature vectors** $d_i \in [0,1]^{|F|}$

Vector Space Model for Content Relevance



Similarity metric:

$$\mathrm{sim}(d_i, q) := \frac{\sum_{j=1}^{|F|} d_{ij}\, q_j}{\sqrt{\sum_{j=1}^{|F|} d_{ij}^2 \ \sum_{j=1}^{|F|} q_j^2}}$$

Documents are **feature vectors** $d_i \in [0,1]^{|F|}$

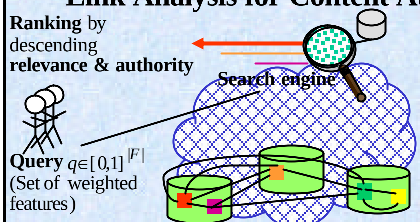
e.g., using:

$$d_{ij} := w_{ij} / \sqrt{\Sigma_k w_{ik}^2}$$

$$w_{ij} := \frac{freq(f_j, d_i)}{\max_k freq(f_k, d_i)} \log \frac{\#docs}{\#docs\ with\ f_i}$$

tf*idf formula

Link Analysis for Content Authority



+ Consider in-degree and out-degree of Web nodes:
Authority Rank (d_i) :=
Stationary visit probability [d_i]
in random walk on the Web

Reconciliation of relevance and authority based on weighted sum (with ad hoc weights)

Dimensions of a Large-Scale Search Engine

- **> 2 Bio. (10**9) Web docs + 1 Bio. News docs**
  > 10 Terabytes raw data
- **> 10 Mio. terms**
  > 2 Terabytes index
- **> 150 Mio. queries per day**
  < 1 sec. average response time
- **< 30 days index freshness**
  > 1000 Web pages per second crawled

High-end server farm:
10 000 Intel servers each with
> 1 GB memory, 2 disks, and
partitioned, mirrored data, distributed across all servers,
plus load balancing of queries, remote administration, etc.

(In-) Effectivity of Web Search Engines

query = „Che

But there is hope:
- **exploit structure**
- **explore neighborhood**
- **start at topic directory**

AltaVista: Fermat's last theore
URL: www-groups
and.ac.uk/~history/I

Northernlight: J. D. Biggins- Publi
Random Walk
http:// www.shef.ac.uk/ ~st1jdb/ bibliog.html

Excite: The Official Web Site of Playboy Lingerie Model Mikki Chernoff http://www.mikkichernoff.com/

Google: ...strong convergence \cite{Chernoff}. \begin{theorem} \label{T1} Let...
http://mpej.unige.ch/mp_arc/p/00-277

Yahoo: Moment-generating Functions; Chernoff's Theorem;
http://www.siam.org/catalog/mcc10/bahadur.htm

Mathsearch: No matches found.

From Observations to Research Avenues

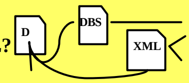
Key observation:
yes, there are ways to find what you are searching,
but **intellectual time is expensive !**
→ requires „intelligent“ automation

Research Avenues:
- Structure and annotate information: **XML**
- Organize documents „semantically“: **ontologies**
- Leverage machine learning: **automatic classification**
- More computer time for better result: **focused crawling**

Goal:
should be able to find results for advanced info request
in one day with < 5 min intellectual effort
that the best human experts can find with infinite time

Challenge: Expert Web Queries

- **Where can I download an open source implementation of the ARIES recovery algorithm?**
- **Find the text and notes of the western song Raw Hide.**
- **What are Chernoff-Hoeffding bounds?**
- **Find Fermat‘s last / Wiles‘ theorem in MathML format.**
- **Are there any theorems isomorphic to my new conjecture? Find related theorems.**
- **Which professors from D are teaching DBS and have research projects on XML?**

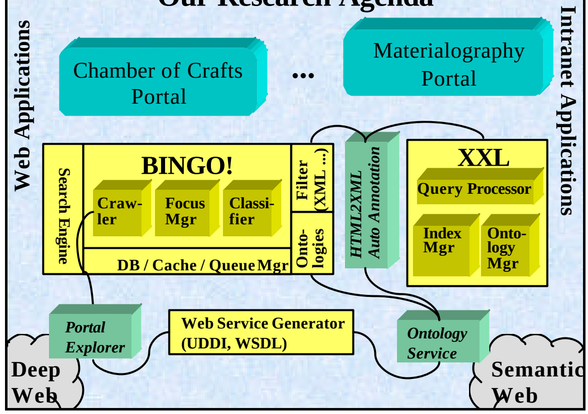


- **Which Shakespeare drama has a scene where a woman talks a Scottish nobleman into murder?**
- **Who was the Italian woman that I met at the PC meeting where Moshe Vardi was PC Chair?**

Challenge: (Meta-) Portal Building

- **Who are the top researchers in the database system community? Who is working on using machine learning techniques for searching XML data?**
- **What are the most important results in large deviation theory?**
- **Find information about public subsidies for plumbers. Find new EU regulations that affect an electrician‘s business.**
- **Which gene expression data from Barrett tissue in the esophagus exhibit high levels of gene A01g ? Are there metabolic models for acid reflux that could be related to the gene expression data?**

Our Research Agenda

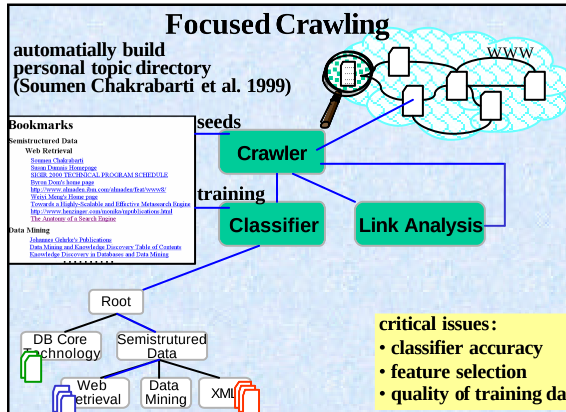


Outline

✓ Performance and QoS Guarantees
✓ Continuous Service Availability
- Intelligent Information Search
  ✓ State of the Art & Research Challenges
  - Focused Crawling

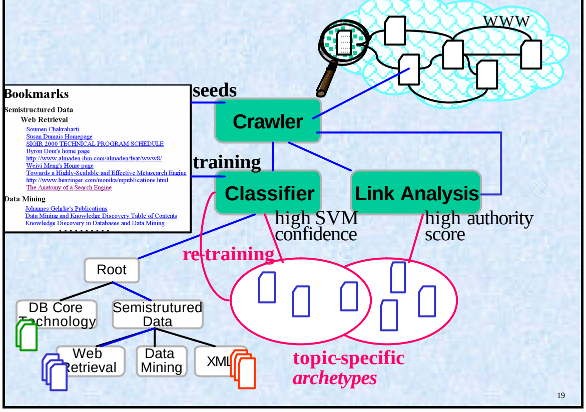
Focused Crawling

automatially build personal topic directory (Soumen Chakrabarti et al. 1999)



critical issues:
- **classifier accuracy**
- **feature selection**
- **quality of training data**

The BINGO! Focused Crawler



BINGO! Adaptive Re-training and Focus Control

for each topic V do {

archetypes(V) := top docs of SVM confidence ranking

È top authorities of V ;

remove from archetypes(V) all docs d that do not satisfy

confidence(d) ³ mean confidence(V) ;

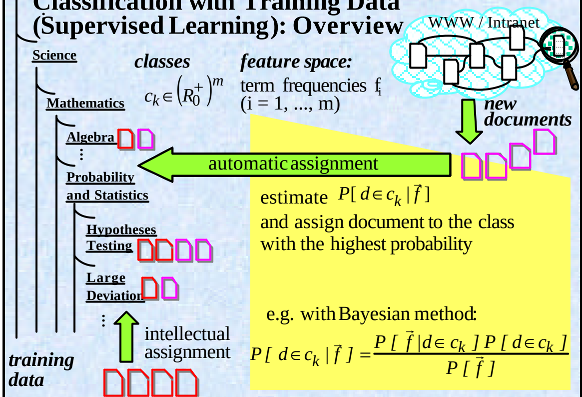
recompute feature selection based on archetypes(V) ;

recompute SVM model for V with archetypes(V) as training data }

combine re-training with two-phase crawling strategy:

- ***learning phase:***
  aims to find archetypes (*high precision*)
  ® hard focus for crawling vicinity of training docs
- ***harvesting phase:***
  aims to find results (*high recall*)
  ® soft focus for long-range exploration with tunnelling

Classification with Training Data (Supervised Learning): Overview



Basic Probability Theory

A **probability space** is a triple (Ω, E, P) with

- a set Ω of elementary events,
- a family E of subsets of Ω with Ω∈E which is closed under ∩, ∪, and − with a countable number of operands (with finite Ω usually E=2^{Ω}), and
- a probability measure P: E → [0,1] with P[Ω]=1 and $P[\cup_i Ai] = \sum_i P[Ai]$ for countably many, pairwise disjoint Ai

Properties of P:

P[A] + P[¬A] = 1 P[∅] = 0

P[A ∪ B] = P[A] + P[B] − P[A ∩ B] P[Ω] = 1

Random Variables

A **random variable** X on the prob. space (Ω, E, P) is a function X: Ω → M with M⊆R s.t. {e | X(e) ≤ x} ∈E for all x ∈M.

F_X: M → [0,1] with $F_X(x)$ = P[X ≤ x] is the ***distribution function*** of X.

With countable set M the function f_X: M → [0,1] with $f_X(x)$ = P[X = x] is called the ***density function*** of X; in general $f_X(x)$ is $F'_X(x)$.

Random variables with countable M are called ***discrete***, otherwise they are called ***continuous***.

For discrete random variables the density function is also referred to as the probability mass function.

Bayes' Theorem

Two events A, B of a prob. space are **independent** if P[A ∩ B] = P[A] P[B].

The **conditional probability P[A | B]** of A under the condition (hypothesis) B is defined as:

$$P[A \mid B] = \frac{P[A \cap B]}{P[B]}$$

Total probability theorem:

For a partitioning of Ω into events B1, ..., Bn:

$$P[A] = \sum_{i=1}^{n} P[A \mid Bi]\, P[Bi]$$

Bayes' Theorem: $P[A \mid B] = \frac{P[B \mid A]\, P[A]}{P[B]}$

a posteriori prob. of A

Naives Bayes Classifier with Bag-of-Words Model

estimate: $P[\,d \in c_k \mid d \text{ hat } \vec{f}\,] \sim P[\,\vec{f} \mid d \in c_k\,]\; P[\,d \in c_k\,]$

with term frequency vector $\vec{f}$

$= \Pi_{i=1}^{m} P[\,f_i \mid d \in c_k\,]\; P[\,d \in c_k\,]$ with feature independence

$= \Pi_{i=1}^{m} \binom{length(d)}{f_i} p_{ik}^{f_i} (1-p_{ik})^{length(d)-f_i} \quad p_k$

with binomial distribution of each feature

or: $= \binom{length(d)}{f_1\, f_2 \ldots f_m} p_{1k}^{f_1}\, p_{2k}^{f_2} \ldots p_{mk}^{f_m} \quad p_k$

with multinomial distribution of feature vectors and

with $\binom{n}{k_1 k_2 \ldots k_m} := \frac{n!}{k_1!\, k_2! \ldots k_m!}$ $\quad \sum_{i=1}^{m} f_i = length(d)$

Example for Naive Bayes

3 classes: c1 – Algebra, c2 – Calculus, c3 – Stochastics
8 terms, 6 training docs d1, ..., d6: 2 for each class

Þ p1=2/6, p2=2/6, p3=2/6

	f1 group	f2 homomorphism	f3 vector	f4 integral	f5 limit	f6 variance	f7 probability	f8 dice
d1:	3	2	0	0	0	0	0	1
d2:	1	2	3	0	0	0	0	0
d3:	0	0	0	3	3	0	0	0
d4:	0	0	1	2	2	0	1	0
d5:	0	0	0	1	1	2	2	0
d6:	1	0	1	0	0	0	2	2

	Algebra k=1	Calculus k=2	Stochastics k=3
p1k	4/12	0	1/12
p2k	4/12	0	0
p3k	3/12	1/12	1/12
p4k	0	5/12	1/12
p5k	0	5/12	1/12
p6k	0	0	2/12
p7k	0	1/12	4/12
p8k	1/12	0	2/12

without smoothing for simple calculation

Example of Naive Bayes (2)

classification of d7: (0 0 1 2 0 0 3 0)

$P[\,\vec{f} \mid d \in c_k\,]\, P[\,d \in c_k\,] = \binom{length(d)}{f_1\, f_2 \ldots f_m} p_{1k}^{f_1}\, p_{2k}^{f_2} \ldots p_{mk}^{f_m} \quad p_k$

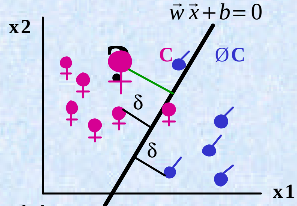
for k=1 (Algebra): $= \binom{6}{1\;2\;3} \left(\frac{3}{12}\right)^1 0^2\, 0^3 \; \frac{2}{6} \qquad = 0$

for k=2 (Calculus): $= \binom{6}{1\;2\;3} \left(\frac{1}{12}\right)^1 \left(\frac{5}{12}\right)^2 \left(\frac{1}{12}\right)^3 \; \frac{2}{6} \qquad = 20 * \frac{25}{12^6}$

for k=3 (Stochastics): $= \binom{6}{1\;2\;3} \left(\frac{1}{12}\right)^1 \left(\frac{1}{12}\right)^2 \left(\frac{4}{12}\right)^3 \; \frac{2}{6} \qquad = 20 * \frac{64}{12^6}$

Result: assign d7 to class C3 (Stochastics)

Classification using Support Vector Machines (SVM)



n training vectors with components (x_1, ..., x_m, C) with C = +1 oder -1

Training:

Compute *separating hyperplane* $\vec{w}\,\vec{x} + b = 0$ that maximizes the min. distance of all positive and negative samples to the hyperplane

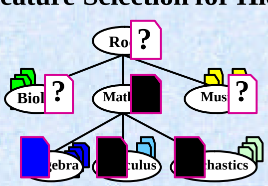
® solve quadratic programming problem

Decision:

Test new vector $\vec{y}$ for membership in C: $(\vec{w}\,\vec{y} + b) = \sum_{i=1}^{m} w_i y_i + b > 0$

Distance of $\vec{y}$ from hyperplane yields classification *confidence*

Feature Selection for Hierarchical Classification



Recursively assign new document to best positively tested topic

Test for topic C_j based on most discriminative features:
select features X_i with highest **mutual information**
(**relative entropy, Kullback-Leibler divergence**)

$$MI(X_i, C_j) = \sum_{X \in \{X_i, \overline{X_i}\}} \; \sum_{C \in \{C_j, \overline{C_j}\}} P[\,X \wedge C\,] \log \frac{P[\,X \wedge C\,]}{P[\,X\,]\, P[\,C\,]}$$

Best features for *Data Mining* (vs. *Web IR* vs. *XML*):
mine, knowledg, OLAP, pattern, discov, cluster, dataset, ...

Feature Space Construction & Meta Strategies

- **possible strategies:**
  - **single term frequencies or tf*idf with top n MI terms**
  - **term pairs within proximity window (e.g., support vector, match about world championship)**
  - **text terms from hyperlink neighbors**
  - **anchor text terms from neighbors (e.g., click here for soccer results)**
- **meta strategies (over m feature spaces for class k):**
  - **unanimous decision:** $C_k(d_j) = 1 \text{ if } \sum_{n=1}^{m} C_k^{(n)} = m$
  - **weighted average:** $C_k(d_j) = 1 \text{ if } \sum_{n=1}^{m} \tilde{p}_k^{(n)} C_k^{(n)} \geq t$
- **strategy n with best ratio of estimated precision to runtime cost**

with x a estimator $\tilde{p}_k^{(n)}$ (Joachims '00) for precision of model n for class k based on leave-one-out training

Link Analysis using Kleinberg's HITS Algorithm

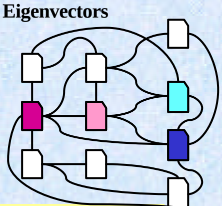
For web graph G=(V,E) and topic-specific base set $B \subseteq V$ find

good authorities with authority score $x_q = \sum_{(p,q)\in E} y_p$

and good hubs with hub score $y_p = \sum_{(p,q)\in E} x_q$

Iterative approximation of principal Eigenvectors

$$\left.\begin{array}{l} \vec{x} = A^T \vec{y} \\ \vec{y} = A\vec{x} \end{array}\right\} \Rightarrow \begin{array}{l} \vec{x} := A^T \vec{y} := A^T A \vec{x} \\ \vec{y} := A\vec{x} := A A^T \vec{y} \end{array}$$



High authority scores indicate good topic representatives

Implementation of the HITS Algorithm

1) Determine sufficient number (e.g. 50-200) of „root pages" via relevance ranking (e.g. using tf*idf ranking)
2) Add all successors of root pages
3) For each root page add up to d successors
4) Compute iteratively the authority and hub scores of this „base set" (of typically 1000-5000 pages) with initialization $x_q := y_p := 1 / |\text{base set}|$ and normalization after each iteration
   → converges to principal Eigenvector (Eigenvector with largest Eigenvalue (in the case of multiplicity 1)
5) Return pages in descending order of authority scores (e.g. the 10 largest elements of vector x)

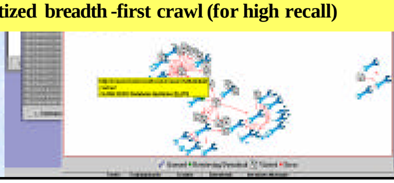
Drawbacks of HITS algorithm:
- relevance ranking within root set is not considered
- susceptible to „topic drift" → extended variants of HITS

Experiment on Information Portal Generation (1)

for single-topic portal on „Database Research"
start with only 2 initial seeds: homepages of DeWitt and Gray
goal: automatic gathering of DBLP author homepages
(with DBLP excluded from crawl)

learning phase for improved feature selection and classification:
depth-first crawl limited to domains of seeds
followed by archetype selection and retraining (for high precision)

harvesting phase for building a rich portal:
prioritized breadth-first crawl (for high recall)



Experiment on Information Portal Generation (2)

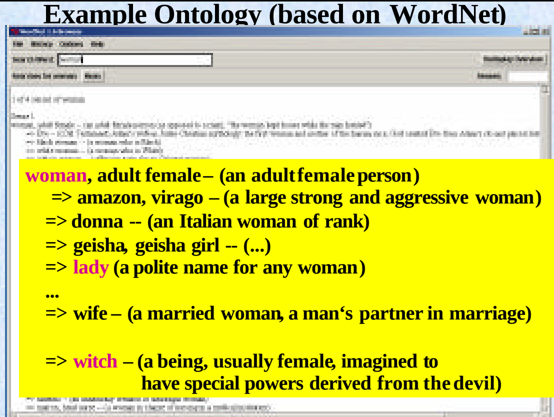
result after 12 hours (on commodity PC):
- 3 mio. URLs crawled on 30 000 hosts, 1 mio. pages analyzed,
- 0.5 mio. pages positively classified
- found 7000 homepages out of 30 000 DBLP authors,
  712 authors of the top 1000 DBLP authors
  with 267 among the 1000 best crawl results

\+ postprocessing for querying and analysis:
- ranking by SVM confidence, authority score, etc.
- clustering, relevance feedback, etc.

Ongoing and Future Work

- Deep Web exploration with auto-generated queries
- Exploiting ontological knowledge
  e.g.: search for a „woman talking someone into murder"
- Construct richer feature spaces
- Exploiting linguistic analysis methods
  e.g.: „... cut his throat ..." — act: killing, subject:..., object:...
- Generalized links & semantic joins, e.g. named entities
- Identifying semantically coherent units
- Combining focused crawling with XML search
  → auto-annotation of HTML, Latex, PDF, etc. docs
  → cross-document querying à la XXL
- User guidance & portal admin methodology
- Exploitation of surf trails from user community

Example Ontology (based on WordNet)



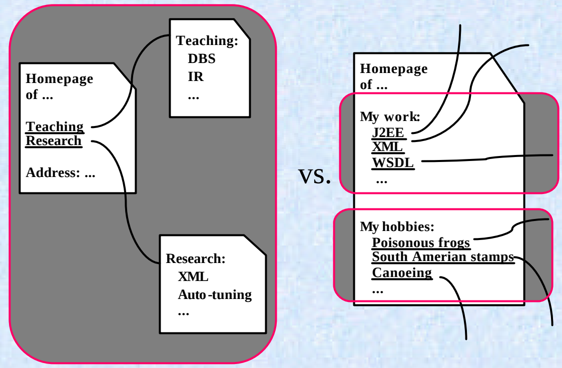
woman, adult female – (an adult female person)
=> amazon, virago – (a large strong and aggressive woman)
=> donna -- (an Italian woman of rank)
=> geisha, geisha girl -- (...)
=> lady (a polite name for any woman)
...
=> wife – (a married woman, a man's partner in marriage)

=> witch – (a being, usually female, imagined to have special powers derived from the devil)

Ongoing and Future Work

- Deep Web exploration with auto-generated queries
- Exploiting ontological knowledge
  e.g.: search for a „woman talking someone into murder"
- Construct richer feature spaces
- Exploiting linguistic analysis methods
  e.g.: „... cut his throat ..." — act: killing, subject:..., object:...
- Generalized links & semantic joins, e.g. named entities
- Identifying semantically coherent units
- Combining focused crawling with XML search
  → auto-annotation of HTML, Latex, PDF, etc. docs
  → cross-document querying à la XXL
- User guidance & portal admin methodology
- Exploitation of surf trails from user community

Towards "Semantically Coherent" Units



Ongoing and Future Work

- Deep Web exploration with auto-generated queries
- Exploiting ontological knowledge
  e.g.: search for a „woman talking someone into murder"
- Construct richer feature spaces
- Exploiting linguistic analysis methods
  e.g.: „... cut his throat ..." — act: killing, subject:..., object:...
- Generalized links & semantic joins, e.g. named entities
- Identifying semantically coherent units
- Combining focused crawling with XML search
  → auto-annotation of HTML, Latex, PDF, etc. docs
  → cross-document querying à la XXL
- User guidance & portal admin methodology
- Exploitation of surf trails from user community

Summary: Strategic Research Avenues

Challenges for next-decade
Web information systems:

- **Self-organizing systems built out of self-tuning components for performance and differentiated QoS guarantees**
- **Trouble-free, continuously available Web services with perfect failure masking to application programs**
- **Intelligent organization and searching of information based on synergy of DB, IR, CL, ML, and AI technologies**
  - ® ***large-scale experiments***
  - ® ***more and better theoretical underpinnings***

Conceivable killer arguments:

Infinite RAM & network bandwidth and zero latency for free

Smarter people don't need a better Web