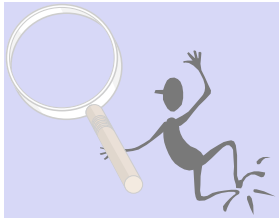
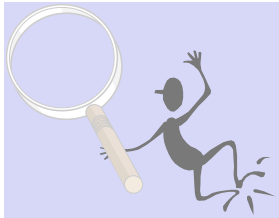


Text and Web Mining

A big challenge for Data Mining

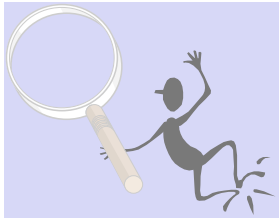
Nguyen Hung Son

Warsaw University

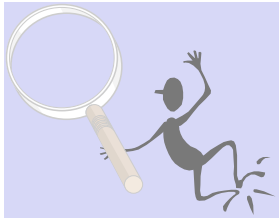


Outline

- **Text vs. Web mining**
- **Search Engine Inside:**
  - **Why Search Engine so important**
  - **Search Engine Architecture**
    - Crawling Subsystem
    - Indexing Subsystem
    - Search Interface
- **Text/web mining tools**
- **Results and Challenges**
- **Future Trends**

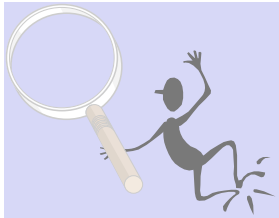


TEXT AND WEB MINING OVERVIEW



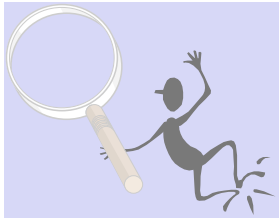
Text Mining

- The sub-domain of Information Retrieval and Natural Language Processing
  - **Text Data**: free-form, unstructured & semi-structured data
  - **Domains**: Internal/intranet & external/internet
    - **Emails, letters, reports, articles, ..**
  - Content management & information organization
  - knowledge discovery: e.g. "*topic detection*", "*phrase extraction*", "*document grouping*", ...



Web mining

- **The sub-domain of IR and multimedia:**
  - **Semi-structured data: hyper-links and html tags**
  - **Multimedia data type: Text, image, audio, video**
  - **Content management/mining as well as usage/traffic mining**



The Problem of Huge Feature Space

A small text database (1.2 MB)

Transaction databases
Typical basket data
(several GBs to TBs)

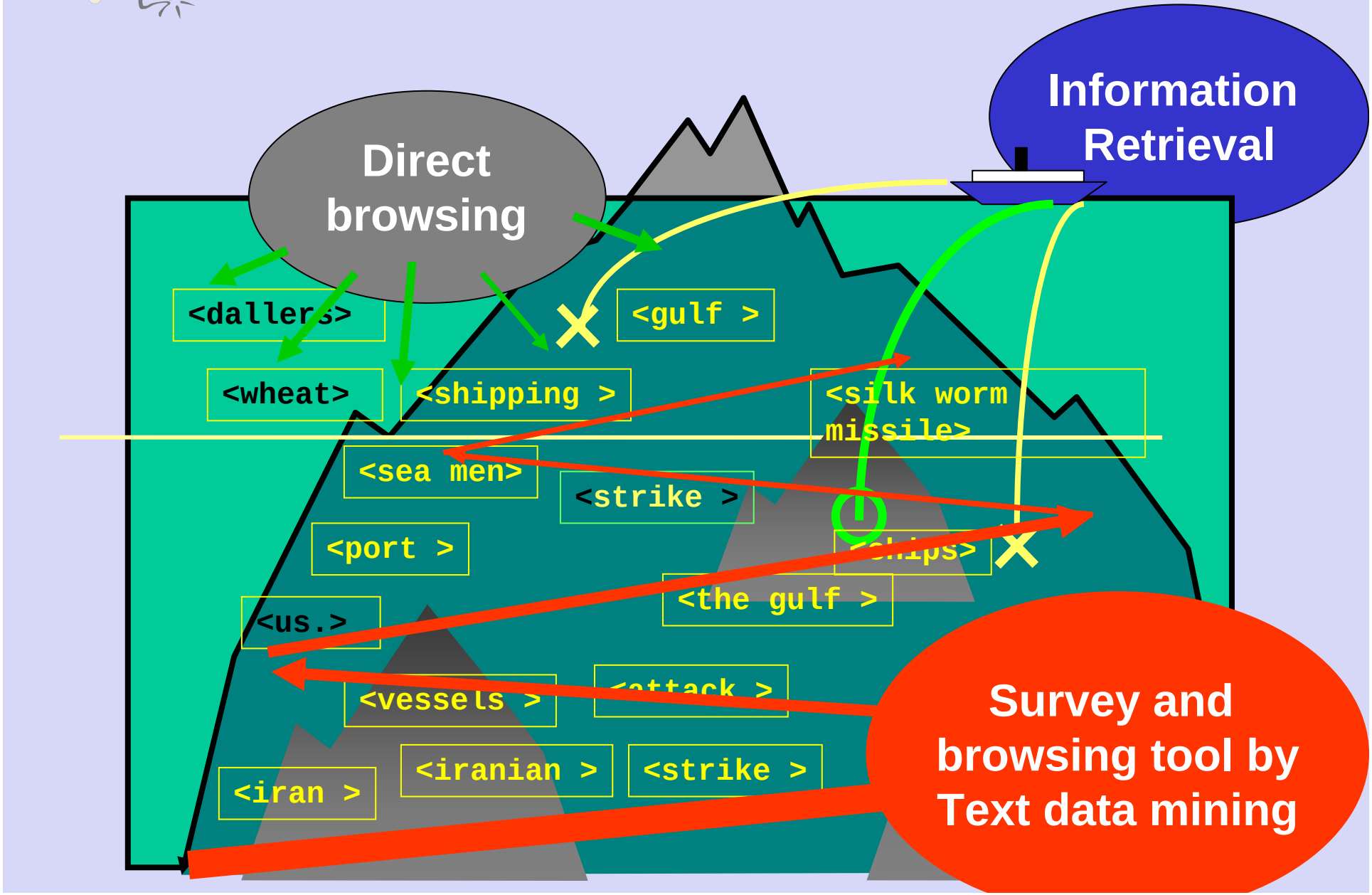
2000 unique items

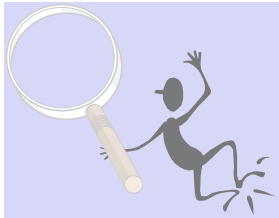
Association rules

700,000 unique phrases!!!

Phrase association patterns

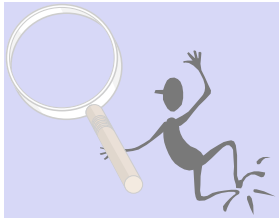
Access methods for Texts





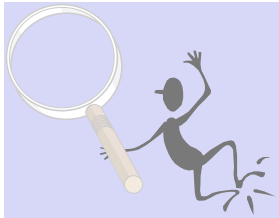
Natural Language Processing

- Text classification/categorization
- Document clustering: finding groups of similar documents
- Information extraction
- Summarization: no corresponding notion in Data Mining



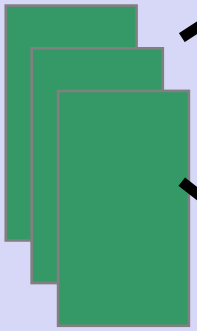
Text Mining vs. NLP

- **Text Mining: extraction of interesting and useful patterns in text data**
  - **NLP technologies as building blocks**
  - **Information discovery as goals**
  - **Learning-based text categorization is the simplest form of text mining**



Text Mining :
Text Refining + Knowledge Distillation

Text collection



Document-based IF

Document Clustering
Text categorization
Visualization
...

Domain dependent

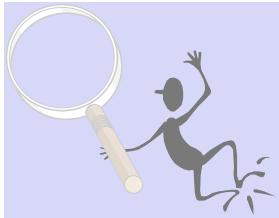
Information Extraction

Concept-based IF

Predictive modelling
Associative discovery
Deviation detection
Trend analysis
...

Domain independent

Text refining ***Intermediate form*** ***Knowledge distillation***

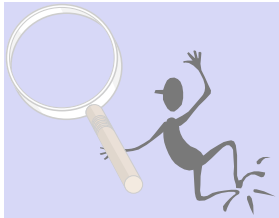


Large Text Databases

- **have emerged** in early 90's **with the rapid progress of communication and network technologies .**
  - **Web pages** (OPENTEXT Index, GBs to TBs)
  - A collection of **SGML documents / XML.**
  - **Genome databases** (GenBank, PIR)
  - **Online dictionary** (Oxford Eng. Dict., 600MB)
  - **Emails** or **plain texts** on a file system.
- **Huge, Heterogeneous, unstructured data**
- **Traditional data mining technology cannot work!**

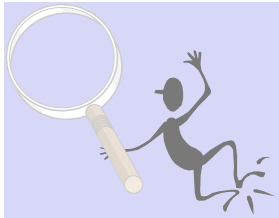
SEARCH ENGINE INSIDE

– ***From Technical Views***



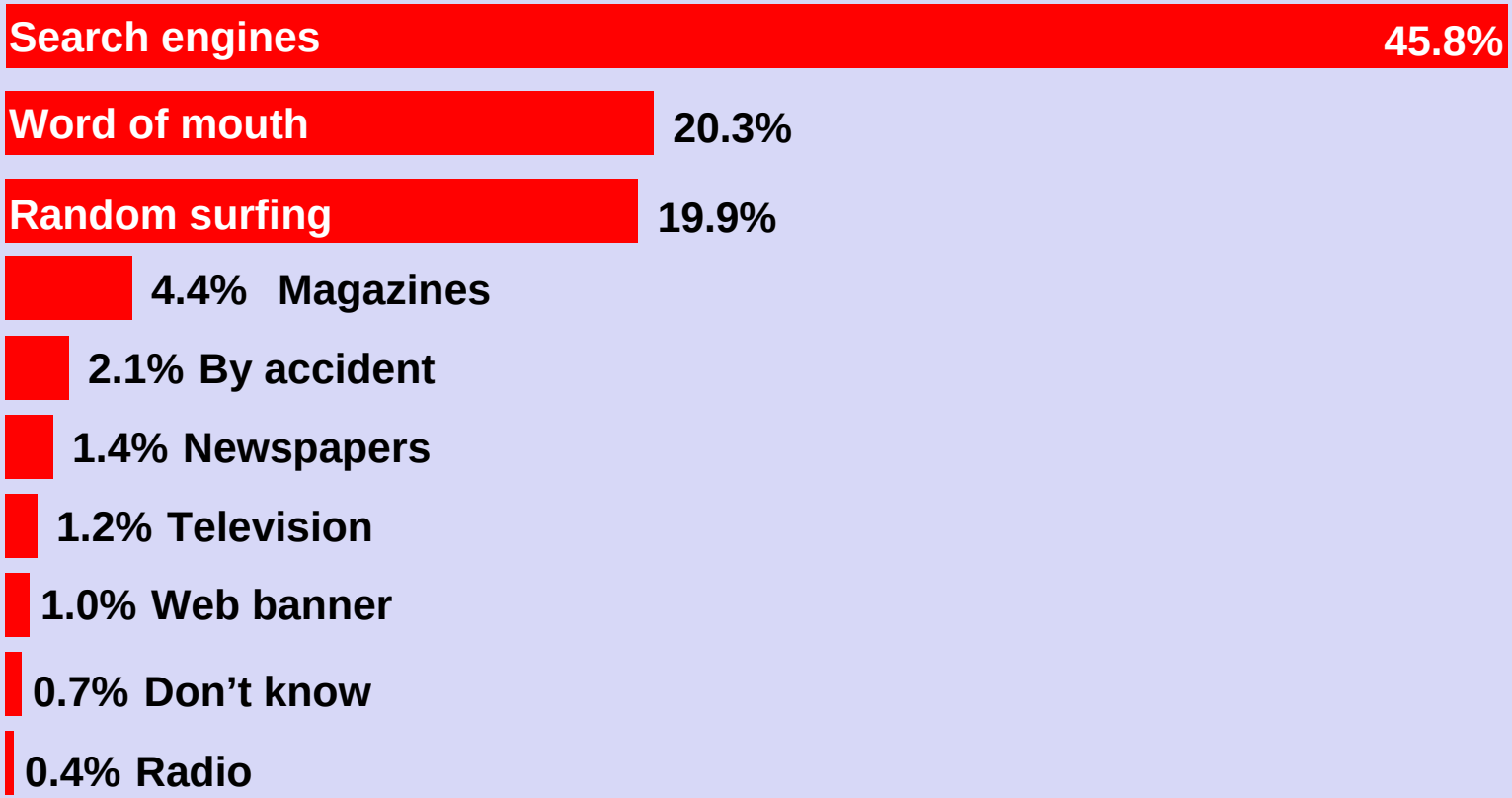
Statistics

- 1 in every 28 page views on the Web is a search result pages. (June 1,1999, Alexa Insider)
- The most widely traveled path on the web in March 1999 was from home.microsoft.com to www.altavista.com . (March 1999, Alexa Insider)
- The average work user spends 73 minutes per month at search engines, second only to 97 minutes at news, info and entertainment sites. (Feb,1999,Internet World)
- Almost 50% of online users turn to search sites for their online news needs. (Dec. 1998, Jupiter)

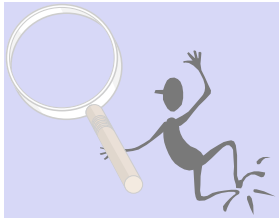


Statistics

How Internet Users Find New Websites



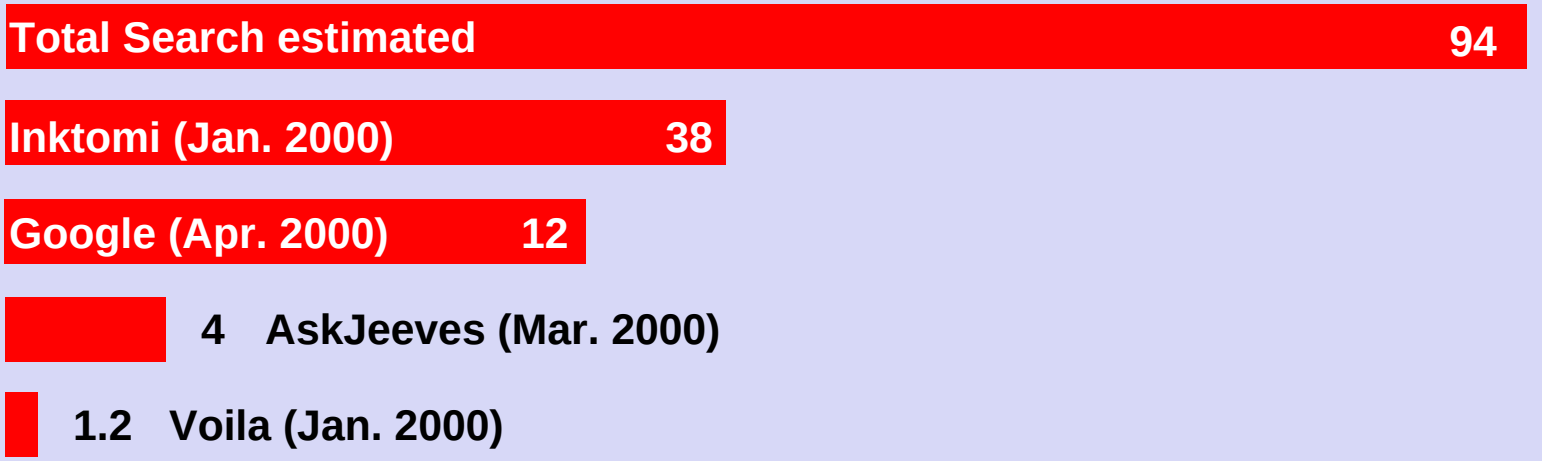
Source: IMP Strategies, Feb, 21. 2000



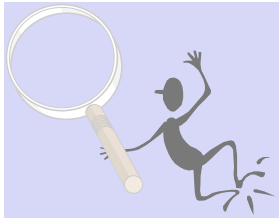
Statistics

Unit : millions/day

How Many Searches are performed

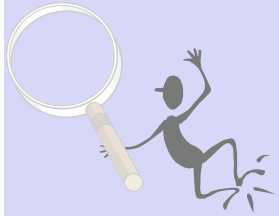


Take Inktomi for example, it should accepts 440 queries each second.



Taxonomy

- General-purpose Search Engine
  `Altavista,Excite,Infoseek,Lycos,HotBot,...`
- Hierarchical Directory
  `Yahoo,Open Directory,LookSmart,...`
- Meta Search Engine
  `MetaCrawler,DogPile,SavvySearch, ...`
- Question-Answering
  `AskJeeves`
- Specialized Search Engines
  `HomePage Finder,Shopping robots,RealName,...`
- `...`



Architecture

Index Manager

Indexer

- Stop list
- Stemmer
- Proper noun list
- Indexing Policy
- Thesaurus
- …

Search Engine

result

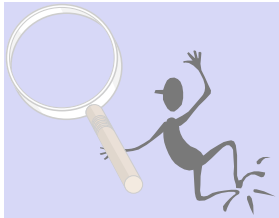
query

Crawler Manager

target url

web content

Spider



Components

- Spider

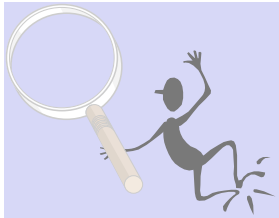
  Spiders crawl the web, collect the documents through what they have found.

- Indexer

  Process and make a logical view of the data.

- Search Interface

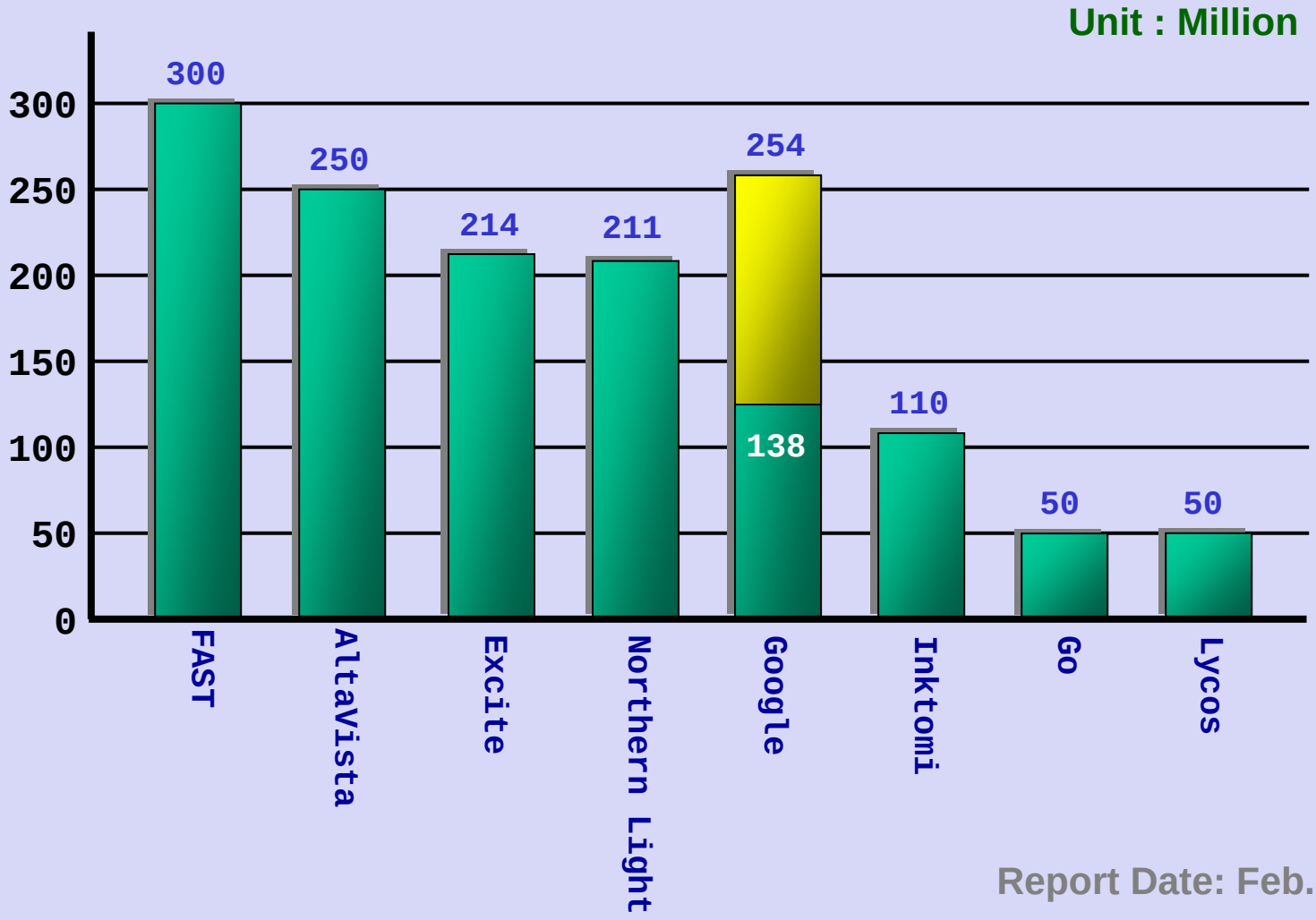
  Accept user queries and search through the index database. Also, rank the result listing and represent to the user.



Crawling Subsystem

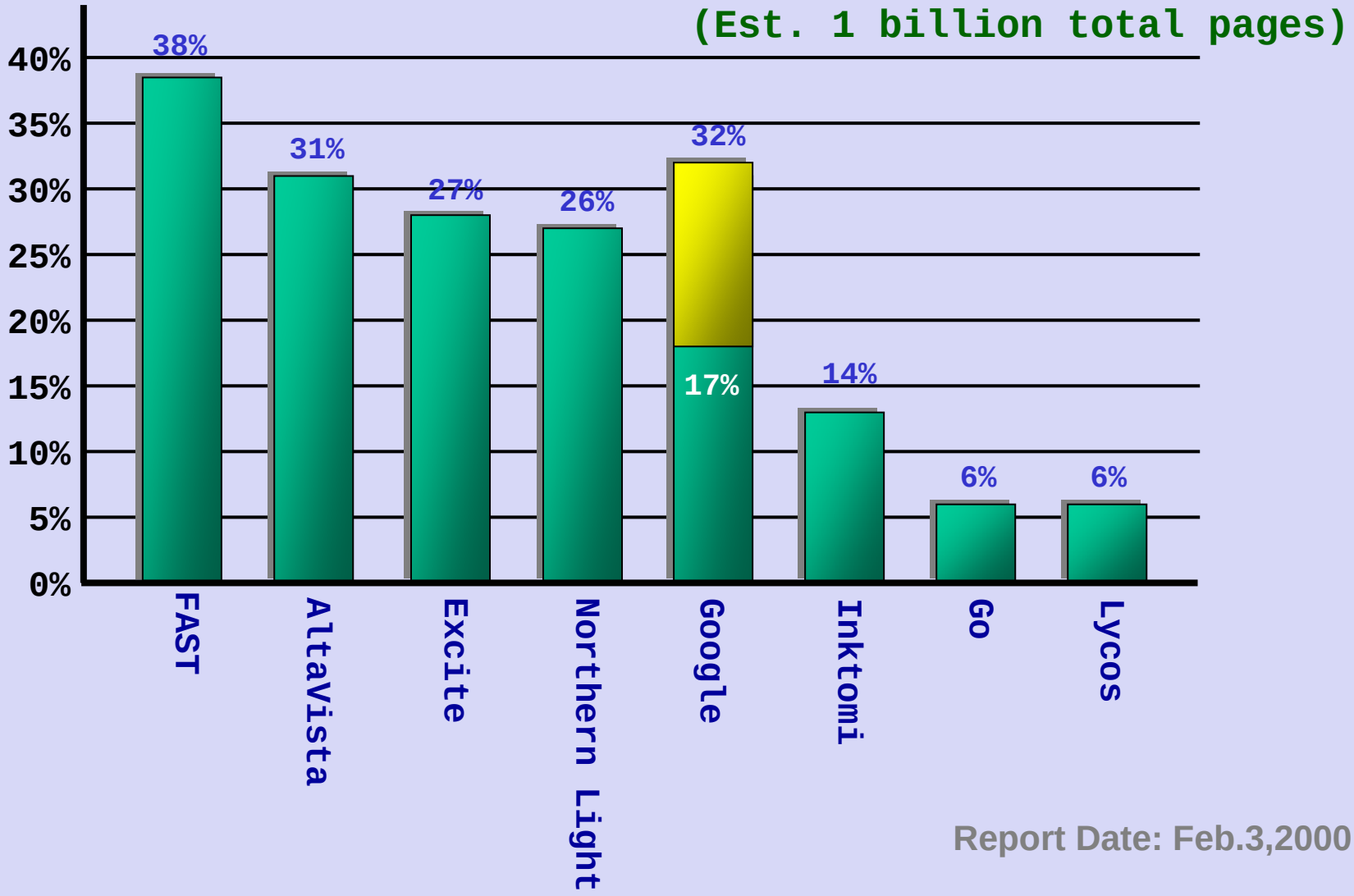
```
Spider (URL)
{
  #Use the HTTP protocol get method to acquire the web page
  Set HttpConnection = HTTPGet(URL);
  #Verify that information is accurate and not a 404 error
  Set Content = CheckInformation(HttpConnection);
  #Place the information into a database for later processing
  StoreInformation(Content);
}
```

Measurement of Indexed Pages

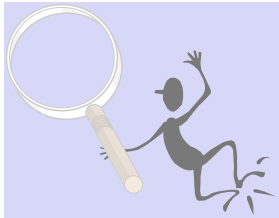


Report Date: Feb.3,2000

Coverage of the Web

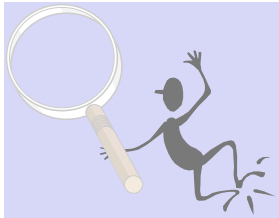


Report Date: Feb.3,2000



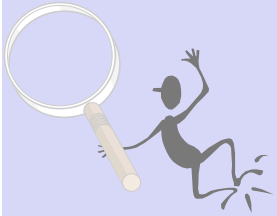
Issues for Crawling (1/3)

- Web Exploration with Priority
  - Decisions about which site(page) is explored first
  - Ensuring document quality and coverage
  - Use Random , BFS, DFS (+depth limits) with priority
- Duplications
  - Host-wise duplications
    - Near 30% of the web are syntactically duplicated
    - ?? are semantically duplicated.
  - Single Host duplications
    - The same website with different host name
    - Symbolic links will cause some infinite routes in the web graph
  - Use Fingerprint, limited-depth exploration
- Dynamic Documents
  - Whether retrieve dynamic documents or not ?
  - Single dynamic document with different parameters ?!



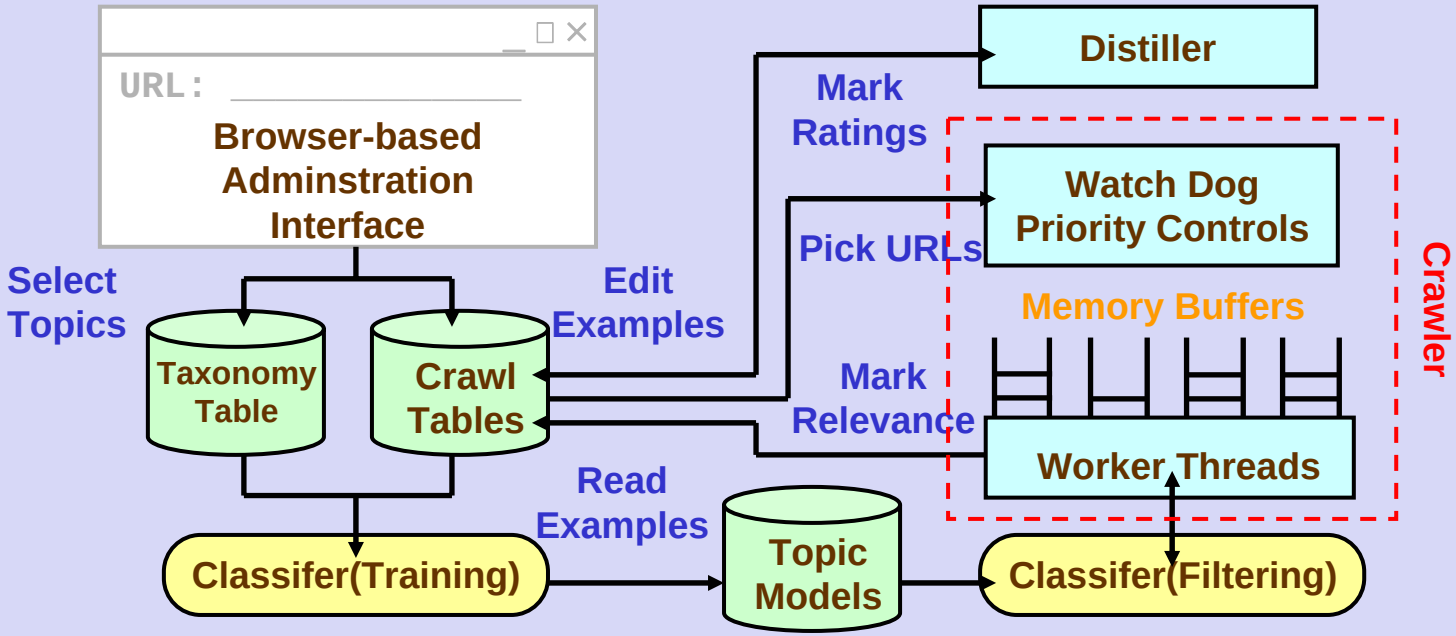
Issues for Crawling (2/3)

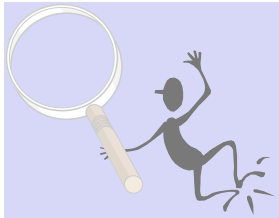
- Load Balance
  - Internal
    - Response time, size of answers are unpredictable
    - There are additional system constraints (# threads,# open connections, etc)
  - External
    - Never overload websites or network links (A well-connected crawler can saturate the entire outside bandwidth of some small country)
    - Support robot standard for politeness.
- Storage Management
  - Huge amount of url/document data
- Freshness
  - Many web sites(pages) changes oftenly, others nearly remains unchanged
  - Revisit different website with different periods.



Issues for Crawling (3/3)

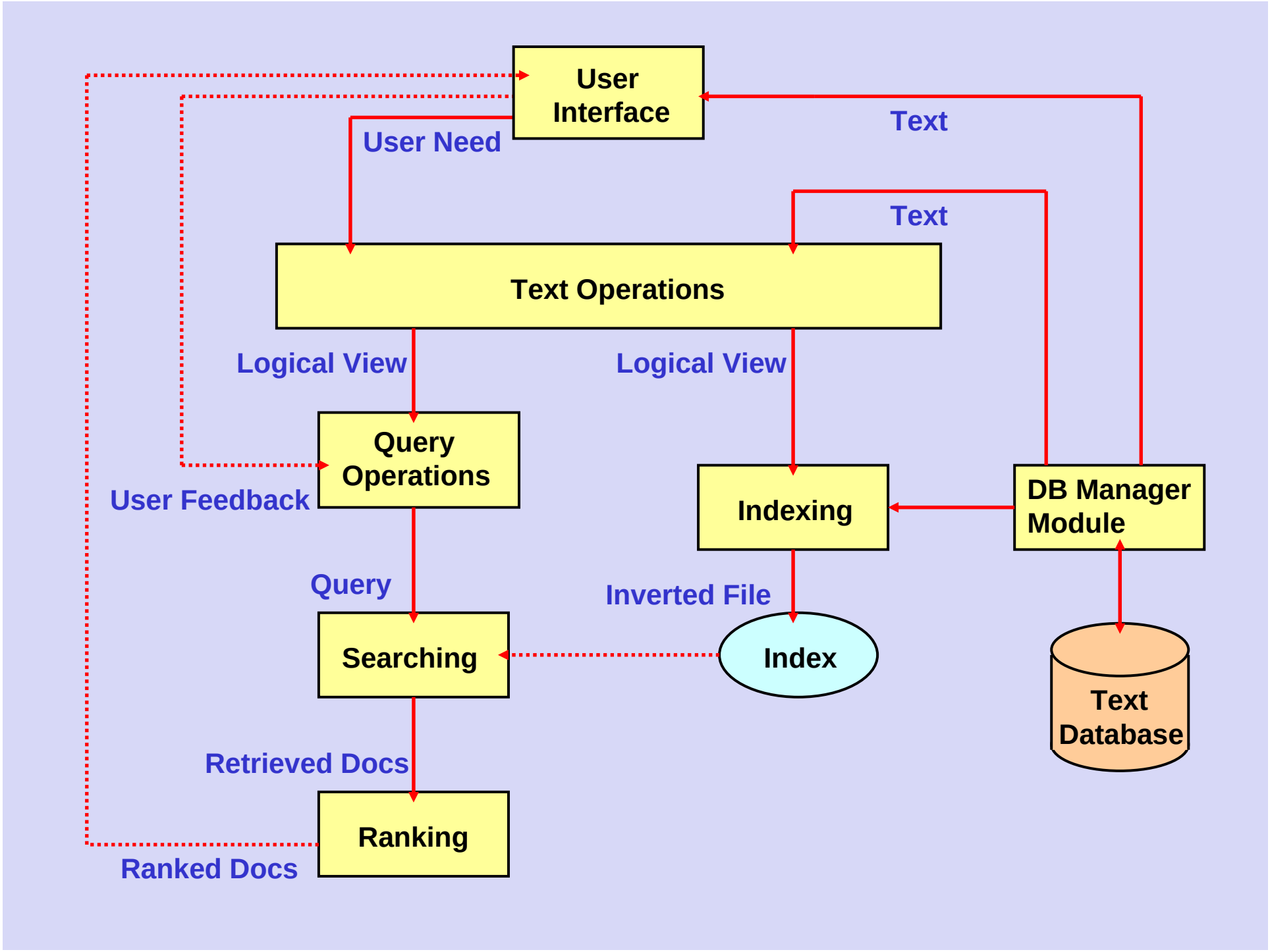
- **The Hidden Web**
  - Some websites are not popular but valuable
  - Use Fast DNS search for possible explorations.
- **Sample Architecture of Crawling System** (Adapted from a topic-specific crawler)

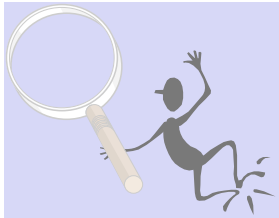




Indexer Subsystem

```
Index(content,URL) {
  #Search each needed HTML structure
  Set Head=GetHtmlHead(content);
  Set Title=GetHtmlTitle(content);
  Set Keywords=GetHtmlKeyword(content);
  #Get needed keywords
  Loop  {
    Set Object = CreateObject(Keywords,Title,Head,URL);
    #Store the keyword, and make internal representation
    StoreKeyword(Object,keyword);
  }
}
```


User Interface
User Need
Text
Text
Text Operations
Logical View
Logical View
Query Operations
User Feedback
Indexing
DB Manager Module
Query
Inverted File
Searching
Index
Text Database
Retrieved Docs
Ranking
Ranked Docs

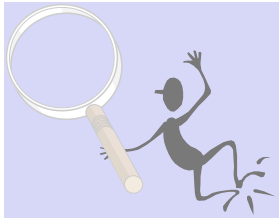


Logic View of Docs and Queries

from Vector Space Model

- Documents and Queries are treated as a t-dimension vectors
  - t is the dimension of the whole index term space.
  - Each vector component is the weight for relevance factor for a specific index term.
- Typical measurement for relevance

$$sim(d_j,q) = \frac{\vec{d_j} \bullet \vec{q}}{|\vec{d_j}| \times |\vec{q}|}$$



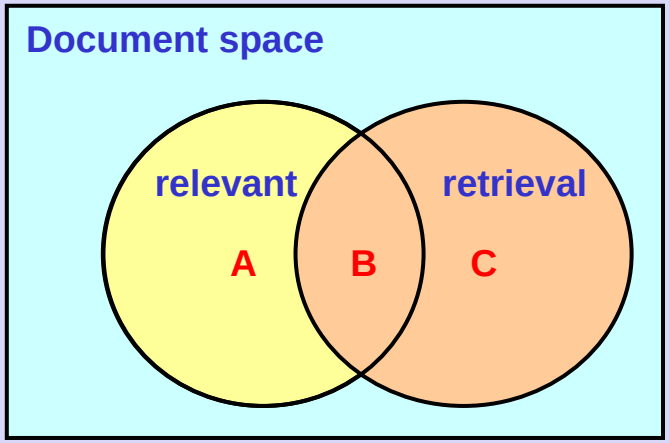
Logic View of Docs and Queries
from Vector Space Model

- **Typical weighting scheme – TFxIDF**

$$W_{i,j} = f_{i,j} \times \log \frac{N}{n_i}$$

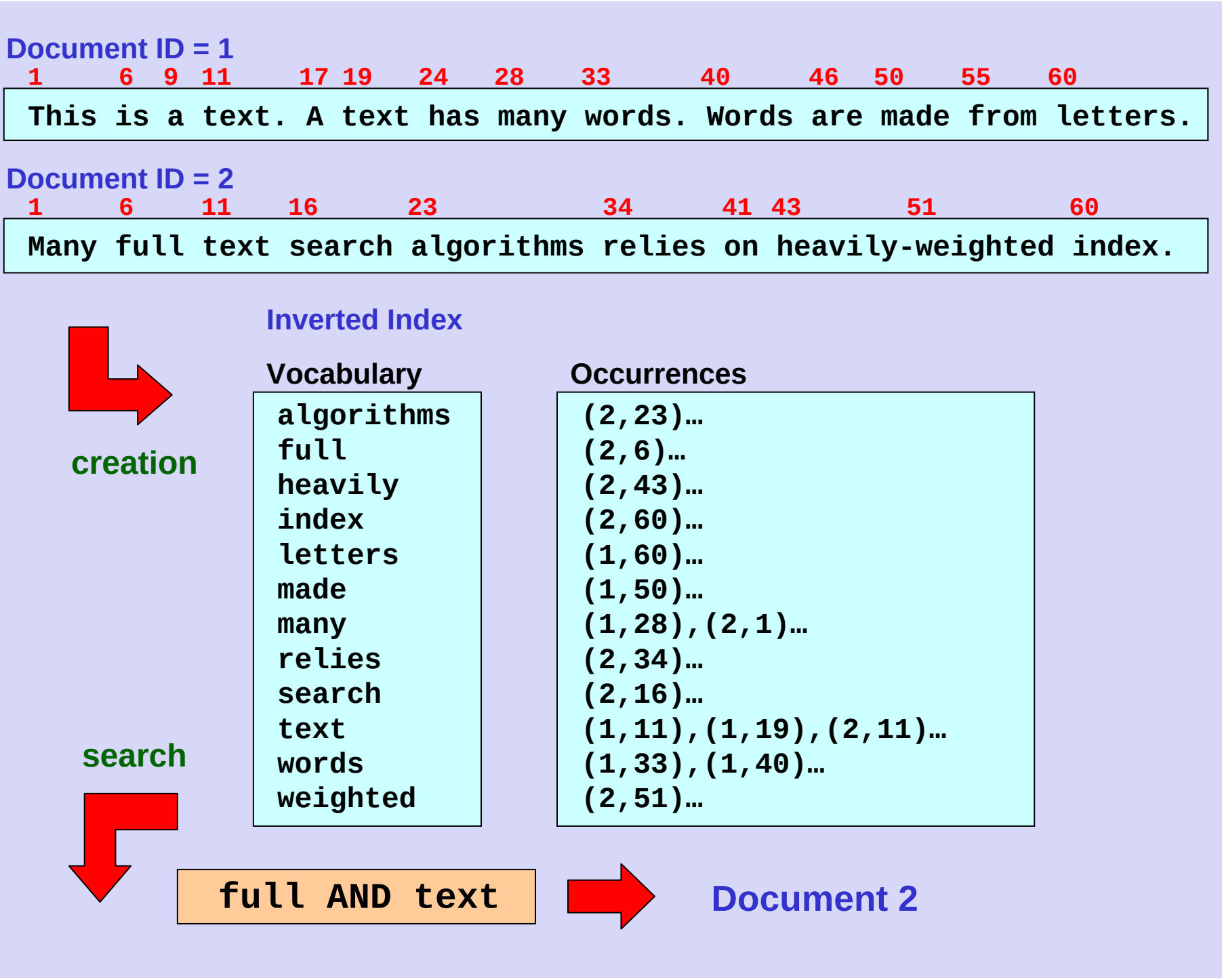
$F_{i,j}$: $term_i$'s frequency in $document_j$
N : total number of documents
N_i : total number of occurrence in different documents

- **Typical Effectiveness Measurement – Recall/Precision**



Recall = the fraction of the relevant documents which has been retrieved

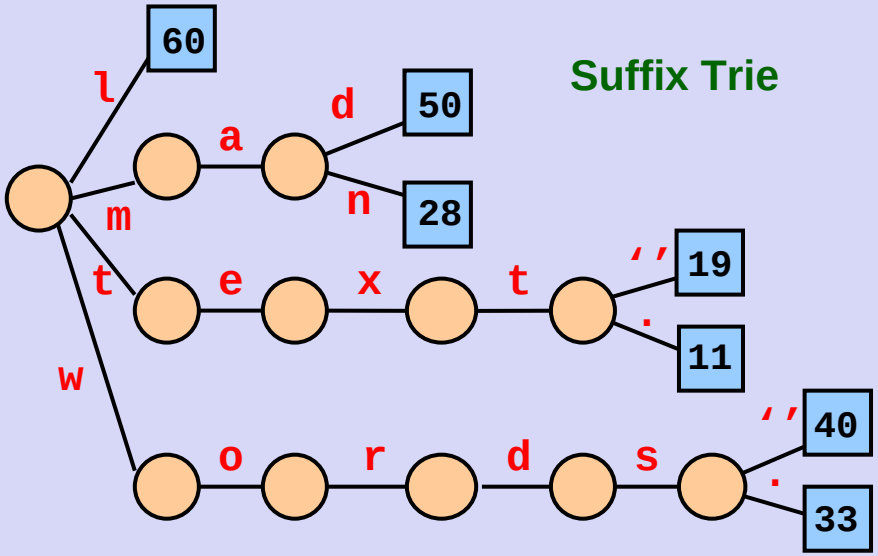
Precision = the fraction of the retrieved documents which is relevant


Document ID = 1
1 6 9 11 17 19 24 28 33 40 46 50 55 60
This is a text. A text has many words. Words are made from letters.
Document ID = 2
1 6 11 16 23 34 41 43 51 60
Many full text search algorithms relies on heavily-weighted index.
Inverted Index
Vocabulary
Occurrences
algorithms
full
heavily
index
letters
made
many
relies
search
text
words
weighted
(2,23)…
(2,6)…
(2,43)…
(2,60)…
(1,60)…
(1,50)…
(1,28),(2,1)…
(2,34)…
(2,16)…
(1,11),(1,19),(2,11)…
(1,33),(1,40)…
(2,51)…
creation
search
full AND text
Document 2

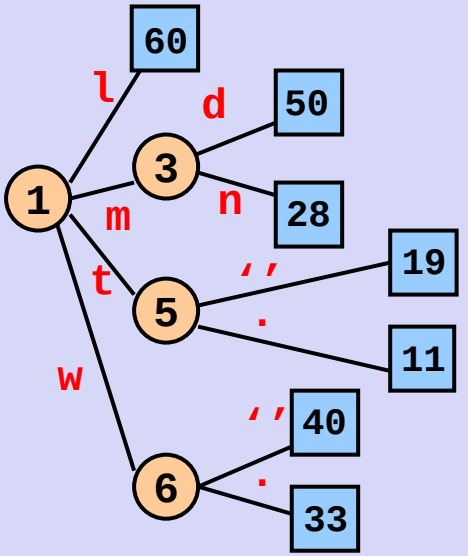
Text

1 6 9 11 17 19 24 28 33 40 46 50 55 60

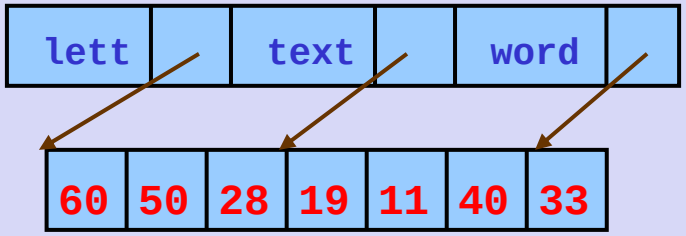
```
This is a text. A text has many words. Words are made from letters.
```



Suffix Trie



Suffix Tree



Supra-Index

Suffix Array

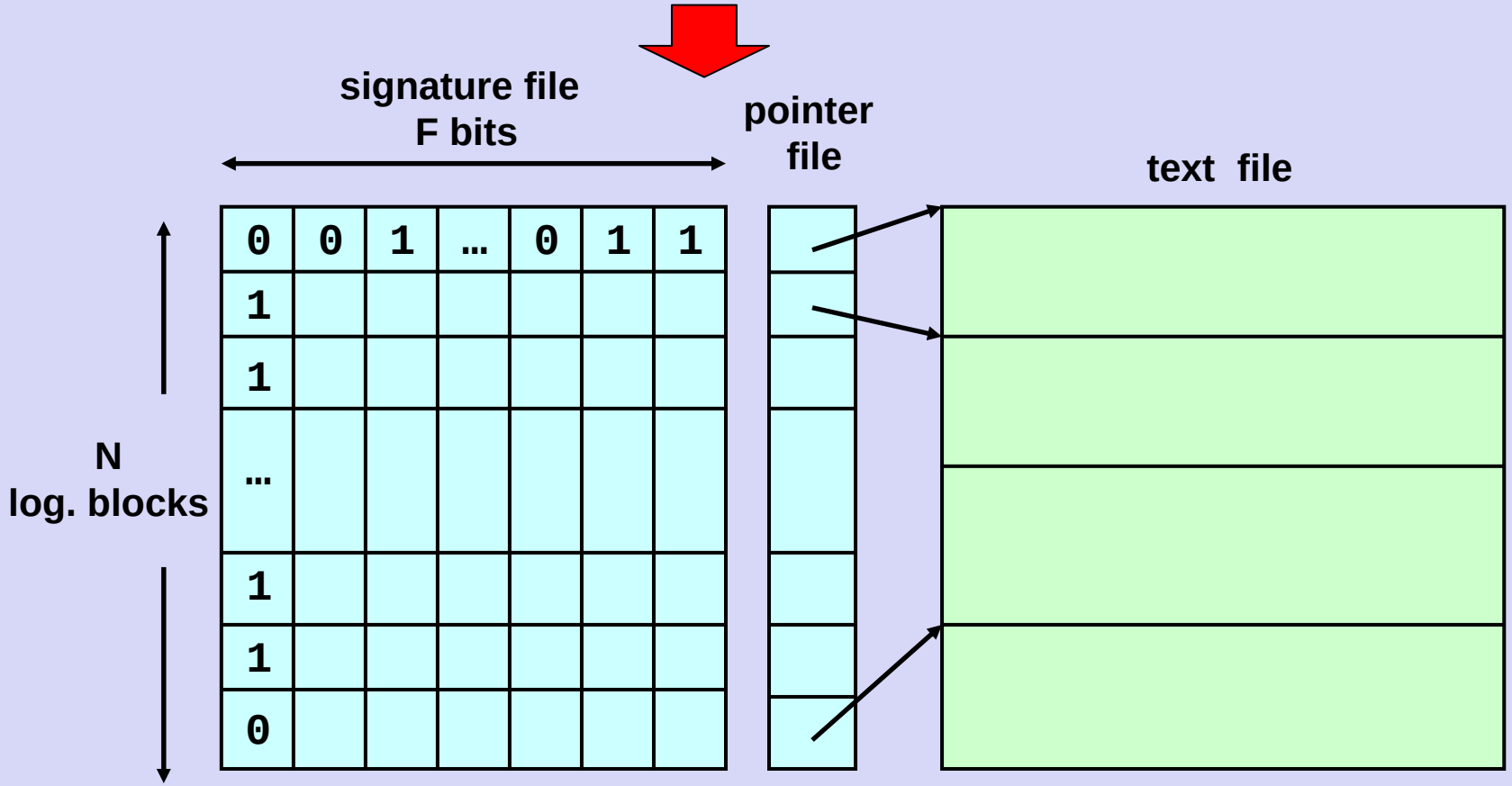
Text

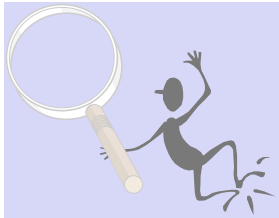
This is a text. A text has many words. Words are made from letters.

Word	Signature
text	001 000 110 010
many	000 010 101 001
Block Signature	001 010 111 011

Parameter

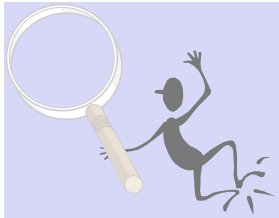
- D logical block
- F signature size in bits
- m number of bits per word
- Fd false drop probability





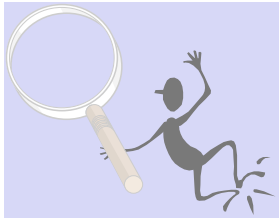
Issues for Indexing (1/2)

- **Language Identification**
  - Documents with different languages should be unified into a meta-representation.
  - Code conversion without concept lose.
  - How to identify language type
    - use meta data (charset, content-encoding) if available.
    - statistical approaches to identify language type
- **Storage Management**
  - Huge amount of indexes can not be loaded in the memory totally
  - Use cache mechanism, fast secondary storage access…
  - Efficient database structures
  - Using Compression ?! Speed and Storage tradeoff



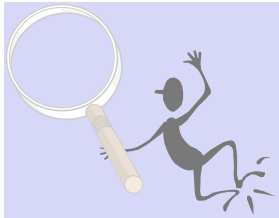
Issues for Indexing (2/2)

- Text Operations
  - Full text or controlled vocabulary
  - Stop list, Stemming, Phrase-level indexing, Thesaurus…
  - Concept discovery, Directory establishment, Categorization
  - Support search with fault tolerances ?!
  - …
- Query-independent ranking
  - Weighting scheme for query-independent ranking
  - Web graph representation manipulations
- Structure information reservation
  - Document author, creation time, title, keywords, …



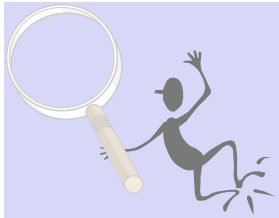
Search Subsystem

```
Report (query) {
  #Get all relevant URLs in the internal database
  Set Candidates = GetRelevantDocuments(query);
  #Rank the lists according to its relevance scores
  Set Answer = Rank(Candidates);
  #Format the result
  DisplayResults();
}
```



What makes Web Users So Different

- **Make poor queries**
  - **Short queries** (2.35 terms for English, 3.4 characters for Chinese)
  - **Imprecise terms**
  - **Sub-optimal syntax** (80% queries without operator)
- **Wide variance in**
  - **Needs** (Some are looking for proper noun only)
  - **Expectations**
  - **Knowledge**
  - **Bandwidth**
- **Specific behavior**
  - **85% look over one result screen only**
  - **78% of queries are not modified**
  - **Follow links**

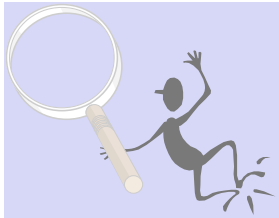


Ranking

- **Goal**

  order the answer set to a query in decreasing order of value

- **Types**
  - **Query-independent : assign an intrinsic value to a document, regardless of the actual query**
  - **Query-dependent : value is determined only with respect to a particular query**
  - **Mixed : combination of both valuations**
- **Examples**
  - **Query-independent : length, vocabulary, publication data, number of citations(indegree), etc**
  - **Query-dependent : cosine measurement**

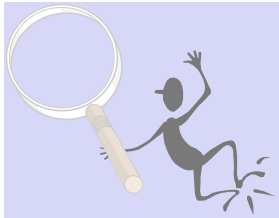


Some ranking criteria

- Content-based techniques

  Variant of term vector model or probabilistic model
- Ad-hoc factors

  Anti-porn heuristics, publication/location data
- Human annotations
- Connectivity-based techniques
  - Query-independent
    - **PageRank [PBMW ’98, BP ’98] , indegree [CK’97] …**
  - Query-dependent
    - **HITS [K’98] …**



Connectivity-Based Ranking

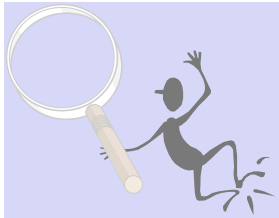
- PageRank
  - Consider a random Web surfer
    - **Jumps to random page with probability α**
    - **With probability 1 - α, follows a random hyperlink**
  - Transition probability matrix is

    $\alpha \times U + (1- \alpha) \times A$

    where U is the uniform distribution and A is adjacency matrix
  - Query-independent rank = stationary probability for this Markov chain

    $PR(a) = \alpha + (1 - \alpha) \sum PR(P_i) / C(P_i)$

  - Crawling the Web using this ordering has been shown to be better than other crawling schemes.



Practical Systems - Altavista

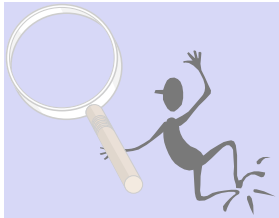
- **Altavista configuration ’98**
  - **Crawler - Scooter**
    - 1.5 GB memory
    - 30 GB RAID disk
    - 4x533 MHz AlphaServer
    - 1 GB/s I/O bandwidth
  - **Indexing Engine – Vista**
    - 2 GB memory
    - 180 GB RAID disk
    - 2x533 MHz AlphaServer
  - **Search Engine – Altavista**
    - 20 multi-processor machines
    - 130 GB memory
    - 500 GB RAID disk

Don’t be surprised about it !!

- Inktomi uses a cluster of hundreds of Sun Sparc workstation with 75 GB RAM, over 1 TB disk.
- It crawls 10 millions pages a day.

How Well does it Perform?

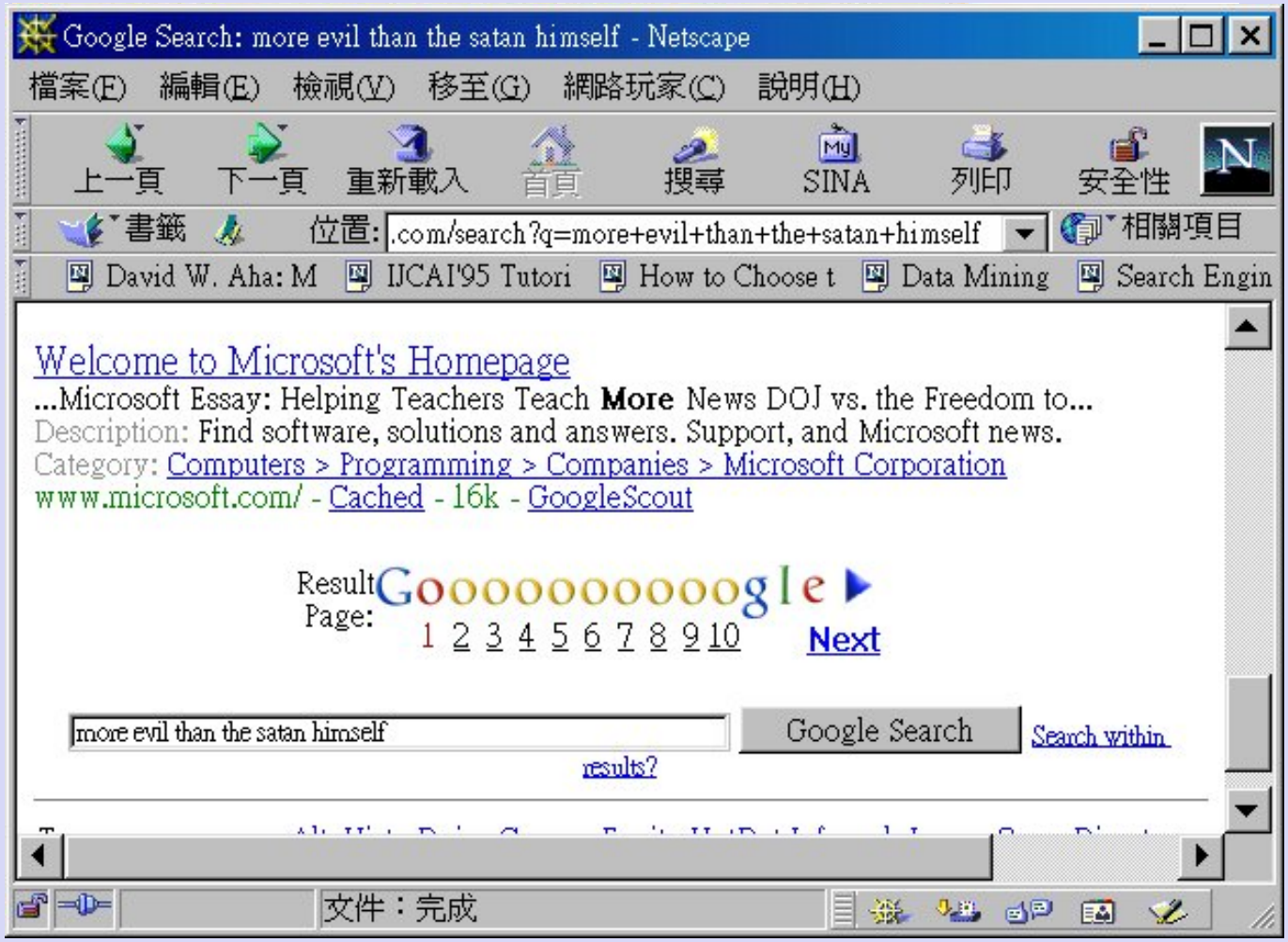
- Index about 0.8TB text
- No stop words
- 37 million queries on weekdays
- Mean response time = 0.6 sec

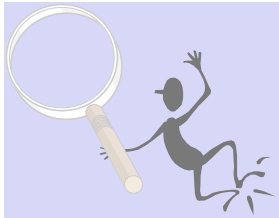


Practical Systems

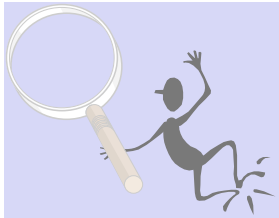
Google!

- The power of PageRank



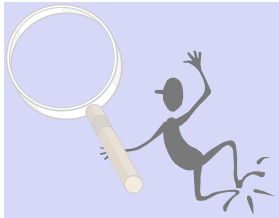


A REVIEW OF SEARCH ENGINE STUDIES AND CHALLENGES



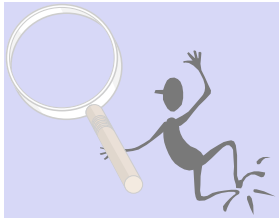
“Intelligent” web agents

- Some intelligent crawlers have built-in learning algorithms:
  - text classification (for domain-specific data base)
  - path finding (using reinforcement learning)
- Some Search Engines use Inference Networks / Belief Networks for document ranking



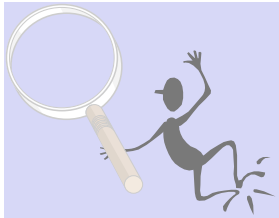
Information Retrieval results

- Measures
  - Recall measures
  - Precision measures
  - Real user evaluation measures????
- Transaction Log Analysis
- Defining Web Searching Studies

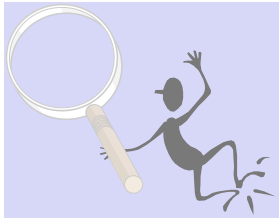


Challenges

- **Comparison framework?**
  - **Descriptive Information**
  - **Analysis Presentation**
    - session
    - query
    - term (phrase)
  - **Statistical analysis**

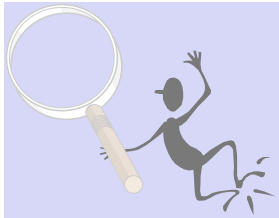


THE FUTURE



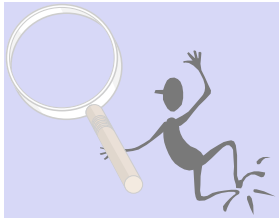
Text/Web Mining - The Market Place

- **Many products/companies**
  - **High tech start-up and big players**
  - **Battle field is in industry rather than in academic institutes.**
- **Functions**
  - **Search & retrieval**
  - **Document navigation & exploration**
  - **Text analysis**
  - **Knowledge management**



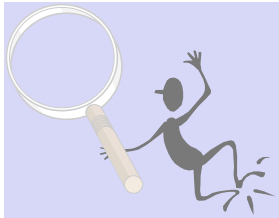
Future Trends

- Multi-lingual/Cross-Lingual Information Retrieval
  - Another way toward concept-oriented searching
- Web Mining
  - Web content mining : customer behavior analysis, advertisment
  - Web usage mining : web query log analysis
- Personalized Search Agents
  - Information filtering, information routing
  - More accurate user concept hierarchy mapping
- Topic-specific knowledge base creation
- Question-Answering system
  - Intelligent e-Service
  - User modeling research



Promising directions

- Content personalization
- Multilingual
- New content/knowledge discovery
- Domain-specific applications
  - Personalized portal
  - Competitive intelligence
  - Knowledge management



References

- KRDL's Text Mining site
  - http://textmining.krdl.org.sg/resources.html
- KDNuggets:Data Mining and Knowledge Discovery Resources
  - http://www.KDNuggets.com
- PRICAI'2000 Workshop on Text and Web Mining, Melbourne, 28 August 2000