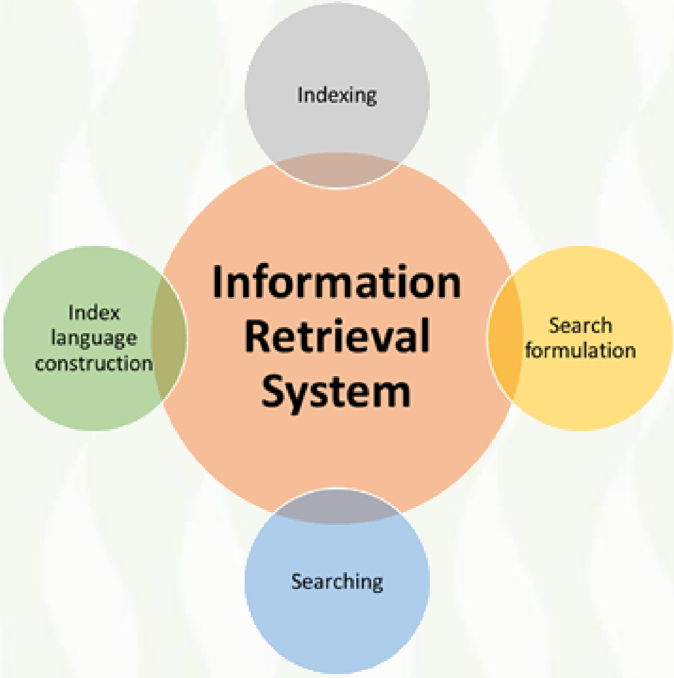
RIYAZ MOHAMMED

INFORMATION RETRIEVAL SYSTEMS

JNTUH/B. TECH/CSE/R18



SYLLABUS

UNIT – I

Introduction to Information Retrieval Systems: Definition of Information Retrieval System, Objectives of Information Retrieval Systems, Functional Overview, Relationship to Database Management Systems, Digital Libraries and Data Warehouses.

Information Retrieval System Capabilities: Search Capabilities, Browse Capabilities, Miscellaneous Capabilities.

UNIT – II

Cataloging and Indexing: History and Objectives of Indexing, Indexing Process, Automatic Indexing, Information Extraction.

Data Structure: Introduction to Data Structure, Stemming Algorithms, Inverted File Structure, N-Gram Data Structures, PAT Data Structure, Signature File Structure, Hypertext and XML Data Structures, Hidden Markov Models.

UNIT – III

Automatic Indexing: Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing, Hypertext Linkages.

Document and Term Clustering: Introduction to Clustering, Thesaurus Generation, Item Clustering, Hierarchy of Clusters.

UNIT – IV

User Search Techniques: Search Statements and Binding, Similarity Measures and Ranking, Relevance Feedback, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the INTERNET and Hypertext.

Information Visualization: Introduction to Information Visualization, Cognition and Perception, Information Visualization Technologies.

UNIT – V

Text Search Algorithms: Introduction to Text Search Techniques, Software Text Search Algorithms, Hardware Text Search Systems.

Multimedia Information Retrieval: Spoken Language Audio Retrieval, Non-Speech Audio Retrieval, Graph Retrieval, Imagery Retrieval, Video Retrieval.

******

UNIT – I

CONTENTS

INTRODUCTION TO INFORMATION RETRIEVAL SYSTEMS (IRS)

DEFINITION OF IRS

- ✓ An Information Retrieval System is a system that is capable of storage retrieval and maintenance of information.
  - Information may be a text (including numeric and date data), images, video and other multimedia objects.
- ✓ Information retrieval is the formal study of efficient and effective ways to extract the right bit of information from a collection.
  - The web is a special case.

What is IR?

- ✓ IR is a branch of applied computer science focusing on the representation, storage, organization, access, and distribution of information.
- ✓ IR involves helping users find information that matches their information needs.

INTRODUCTION TO IRS

- ✓ An Information Retrieval System is a system that is capable of storage, retrieval, and maintenance of information.

- ✓ Information in this context can be composed of text (including numeric and date data), images, audio, video and other multi-media objects.
- ✓ Although the form of an object in an Information Retrieval System is diverse, the text aspect has been the only data type that lent itself to full functional processing.
- ✓ The other data types have been treated as highly informative sources, but are primarily linked for retrieval based upon search of the text.
- ✓ An Information Retrieval System consists of a software program that facilitates a user in finding the information file user needs.
- ✓ The system may use standard computer hardware or specialized hardware to support the search subfunction.
- ✓ The first Information Retrieval Systems originated with the need to organize information in central repositories (e.g., libraries) (Hyman-82).

OBJECTIVES OF IRS

- ✓ The general objective of an Information Retrieval System is to minimize the overhead of a user locating needed information.
- ✓ Overhead can be expressed as the time a user spends in all of the steps leading to reading an item containing the needed information (e.g., query generation, query execution, scanning results of query to select items to read, reading non-relevant items).
- ✓ The success of an information system is very subjective, based upon what information is needed and the willingness of a user to accept overhead.
- ✓ Under some circumstances, needed information can be defined as all information that is in the system that relates to a user's need.
- ✓ In other cases it may be defined as sufficient information in the system to complete a task, allowing for missed data. For example, a financial advisor recommending a billion dollar purchase of another company needs to be

sure that all relevant, information on the target company has been located and reviewed significant in writing the recommendation.

✓ A system that supports reasonable retrieval requires fewer features than one which requires comprehensive retrieval.

✓ The two major measures commonly associated with information systems are precision and recall.

✓ When a user decides to issue a search looking for information oil a topic, the total database is logically divided into four segments shown in Figure 1.1.

✓ Relevant items are those documents that contain information that helps the searcher in answering his question.

✓ Non-relevant items are those items that do not provide any directly useful information.

✓ There are two possibilities with respect to each item: it can be retrieved or not retrieved by the user's query. Precision and recall are defined as:

$$\text{Precision} = \frac{Number\_\mathrm{Re}\,trieved\_\mathrm{Re}\,levant}{Number\_Total\_\mathrm{Re}\,trieved}$$

$$\text{Recall} = \frac{Number\_\mathrm{Re}\,trieved\_\mathrm{Re}\,levant}{Number\_Possible\_\mathrm{Re}\,levant}$$

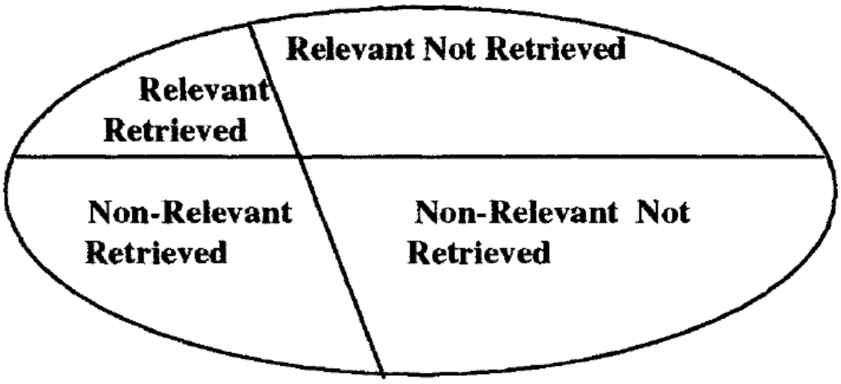


Figure 1.1 Effects of Search on Total Document Space

- ✓ In particular the search tools must assist the user automatically and through system interaction in developing a search specification that represents the need of the user and the writing style of diverse authors (see Figure 1.3).

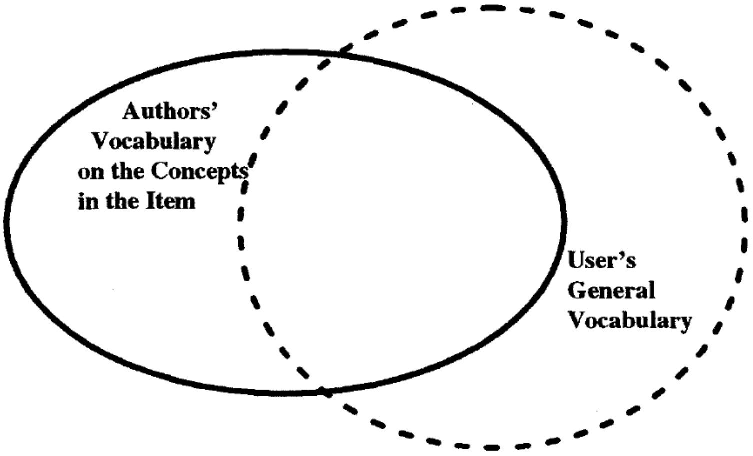

Authors' Vocabulary on the Concepts in the Item
User's General Vocabulary

Figure 1.3 Vocabulary Domains

- ✓ In addition to finding the information relevant to a user's needs, an objective of an information system is to present the search results in a format that facilitates the user in determining relevant items.
- ✓ The new Information Retrieval Systems provide functions that provide the results of a query in order of potential relevance to the user.

FUNCTIONAL OVERVIEW OF IRS

A total Information Storage and Retrieval System is composed of four major functional processes:

1. Item Normalization.
2. Selective Dissemination of Information (i.e., "Mail").
3. Archival Document Database Search.
4. Index Database Search along with the Automatic File Build process that supports Index Files.

Figure 1.4 shows the logical view of these capabilities in a single integrated Information Retrieval System. Boxes are used in the diagram to represent functions while disks represent data storage.

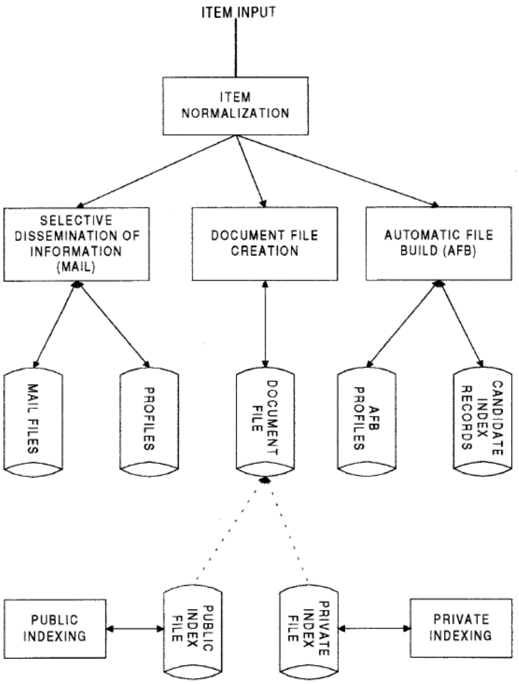


Figure 1.4 Total Information Retrieval System

1. Item Normalization

- ✓ The first step in any integrated system is to normalize the incoming items to a standard format.

- ✓ In addition to translating multiple external formats that might be received into a single consistent data structure that can be manipulated by the functional processes, item normalization provides logical restructuring of the item.
- ✓ Additional operations during item normalization are needed to create a searchable data structure: identification of processing tokens (e.g., words), characterization of the tokens, and stemming (e.g., removing word endings) of the tokens.
- ✓ The original item or any of its logical subdivisions is available for the user to display.
- ✓ The processing tokens and their characterization are used to define the searchable text from the total received text.
- ✓ Figure 1.5 shows the normalization process.

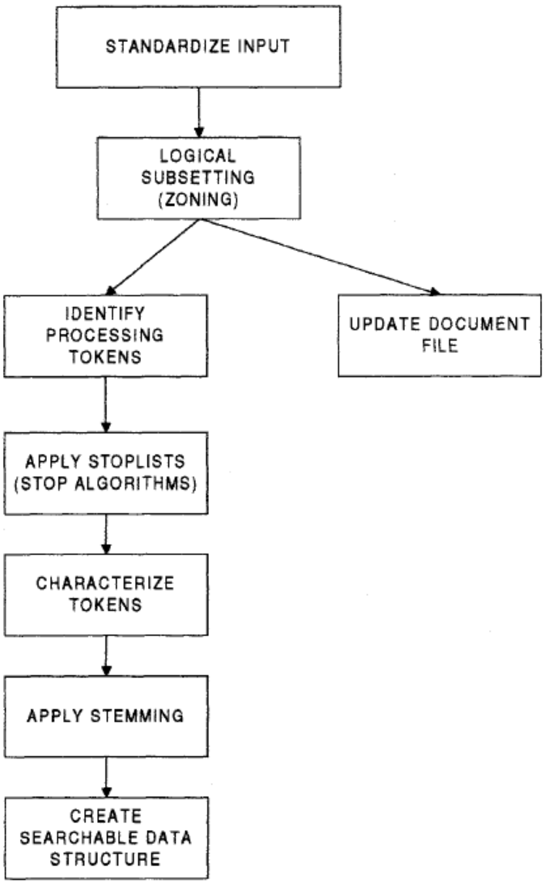

STANDARDIZE INPUT
LOGICAL SUBSETTING (ZONING)
IDENTIFY PROCESSING TOKENS
UPDATE DOCUMENT FILE
APPLY STOPLISTS (STOP ALGORITHMS)
CHARACTERIZE TOKENS
APPLY STEMMING
CREATE SEARCHABLE DATA STRUCTURE

Figure 1.5 The Normalization Process

2. Selective Dissemination of Information

- ✓ The Selective Dissemination of Information (Mail) Process (see Figure 1.4) provides tile capability to dynamically compare newly received items in the information system against standing statements of interest of users and deliver the item to those users whose statement of interest matches the contents of the item.
- ✓ The Mail process is composed of the search process, user statements of interest (Profiles) and user mail files. As each item is received, it is processed against every user's profile.
- ✓ A profile contains a typically broad search statement along with a list of user mail files that will receive the document if the search statement in the profile is satisfied.
- ✓ User search profiles are different than ad hoc queries in that they contain significantly more search terms (10 to 100 times more terms) and cover a wader range of interests.
- ✓ These profiles define all the areas in which a user is interested versus an ad hoc query which is frequently focused to answer a specific question.
- ✓ It has been shown in recent studies that automatically expanded user profiles perform significantly better than human generated profiles (Harman- 95).

3. Document Database Search

- ✓ The Document Database Search Process (see Figure 1.4) provides the capability for a query to search against all items received by the system.
- ✓ The Document Database Search process is composed of the search process, user entered queries (typically ad hoc queries) and the document database which contains all items that have been received, processed and stored by the system.
- ✓ It is the retrospective search source for the system.

- ✓ If the user is on-line, the Selective Dissemination of Information system delivers to the user items of interest as soon as they are processed into the system.
- ✓ Any search for information that has already been processed into the system can be considered a "retrospective" search for information.
- ✓ This does not preclude the search to have search statements constraining it to items received in the last few hours.
- ✓ But typically the searches span far greater time periods. Each query is processed against the total document database.
- ✓ Queries differ from profiles in that they are typically short and focused on a specific area of interest.
- ✓ The Document Database can be very large, hundreds of millions of items or more.
- ✓ Typically items in the Document Database do not change (i.e., are not edited) once received.
- ✓ The value of much information quickly decreases over time.

4. Index Database Search

- ✓ When an item is determined to be of interest, a user may want to save it for future reference. This is in effect filing it. In an information system this is accomplished via the index process.
- ✓ In this process the user can logically store an item in a file along with additional index terms and descriptive text the user wants to associate with the item.
- ✓ It is also possible to have index records that do not reference an item, but contain all the substantive information in the index itself.
- ✓ In this case the user is reading items and extracting the information of interest, never needing to go back to the original item.

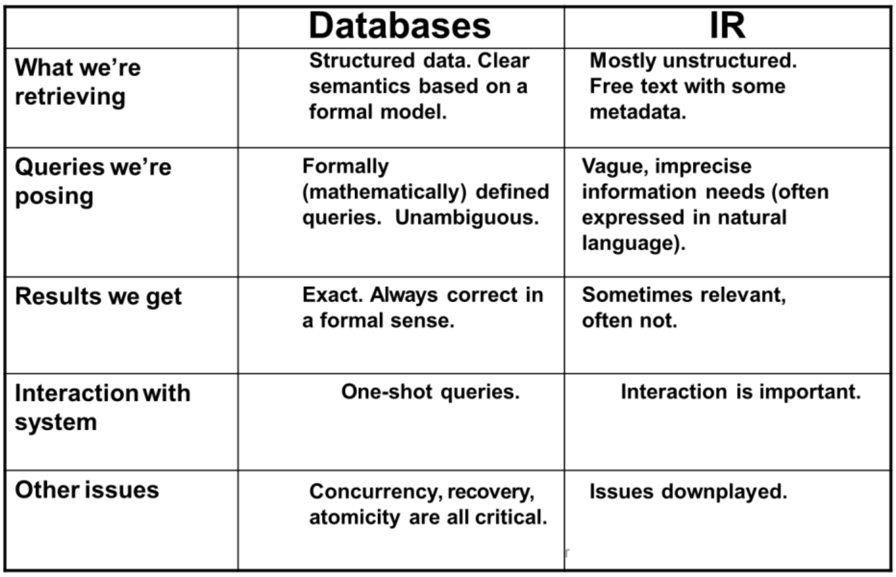
- ✓ The Index Database Search Process (see Figure 1.4) provides the capability to create indexes and search them.
- ✓ The user may search the index and retrieve the index and/or the document it references.
- ✓ The system also provides the capability to search the index and then search the items referenced by the index records that satisfied the index portion of the query. This is called a combined file search.
- ✓ In an ideal system the index record could reference portions of items versus the total item.
- ✓ There are two classes of index files: Public and Private Index files.

<u>RELATIONSHIP TO DATABASE MANAGEMENT SYSTEMS</u>

- ✓ There are two major categories of systems available to process items: Information Retrieval Systems (IRS) and Data Base Management Systems (DBMS).
- ✓ Confusion can arise when the software systems supporting each of these applications get confused with the data they are manipulating.
- ✓ An Information Retrieval System is software that has the features and functions required to manipulate "information" items versus a DBMS that is optimized to handle "structured" data. Information is fuzzy text.
- ✓ The term "fuzzy" is used to imply the results from the minimal standards or controls on the creators of the text items.
- ✓ There is a semantic description associated with each attribute within a table that well defines that attribute.
- ✓ For example, there is no confusion between the meaning of "employee name" or "employee salary" mid what values to enter in a specific database record.
- ✓ On the other hand, if two different people generate an abstract for the same item, they can be different.

- ✓ One abstract may generally discuss the most important topic in an item. Another abstract, using a different vocabulary, may specify the details of many topics.
- ✓ It is this diversity and ambiguity of language that causes the fuzzy nature to be associated with information items.
- ✓ The differences in the characteristics of the data is one reason for the major differences in functions required for the two classes of systems.
- ✓ From a practical standpoint, the integration of DBMS's and Information Retrieval Systems is very important.
- ✓ Commercial database companies have already integrated the two types of systems.
- ✓ One of the first commercial databases to integrate the two systems into a single view is the INQUIRE DBMS.
- ✓ This has been available for over fifteen years.
- ✓ A more current example is the ORACLE DBMS that now offers an imbedded capability called CONVECTIS, which is an informational retrieval system that uses a comprehensive thesaurus which provides the basis to generate "themes" for a particular item.

Databases vs. IR



	Databases	IR
What we're retrieving	Structured data. Clear semantics based on a formal model.	Mostly unstructured. Free text with some metadata.
Queries we're posing	Formally (mathematically) defined queries. Unambiguous.	Vague, imprecise information needs (often expressed in natural language).
Results we get	Exact. Always correct in a formal sense.	Sometimes relevant, often not.
Interaction with system	One-shot queries.	Interaction is important.
Other issues	Concurrency, recovery, atomicity are all critical.	Issues downplayed.

DIGITAL LIBRARIES

- ✓ Two other systems frequently described in the context of information retrieval are Digital Libraries and Data Warehouses (or DataMarts).
- ✓ There is significant overlap between these two systems and an Information Storage and Retrieval System.
- ✓ All three systems are repositories of information and their primary goal is to satisfy user information needs.
- ✓ Information retrieval easily dates back to Vannevar Bush's 1945 article on thinking (Bush-45) that set the stage for many concepts in this area.
- ✓ Libraries have been in existence since the beginning of writing and have served as a repository of the intellectual wealth of society.
- ✓ As such, libraries have always been concerned with storing and retrieving information in the media it is created on.
- ✓ As the quantities of information grew exponentially, libraries were forced to make maximum use of electronic tools to facilitate the storage and retrieval process.
- ✓ With the worldwide interneting of libraries and information sources (e.g., publishers, news agencies, wire services, radio broadcasts) via the Internet, more focus has been on the concept of an electronic library.
- ✓ Between 1991 and 1993 significant interest was placed on this area because of the interest in U.S.
- ✓ Government and private funding for making more information available in digital form (Fox-93).
- ✓ During this time the terminology evolved from electronic libraries to digital libraries.
- ✓ As the Internet continued its exponential growth and project funding became available, the topic of Digital Libraries has grown.

- ✓ By 1995 enough research and pilot efforts had started to support the 1st ACM International Conference on Digital Libraries (Fox-96).
- ✓ There remain significant discussions on what is a digital library.
- ✓ Everyone starts with the metaphor of the traditional library.
- ✓ The question is how do the traditional library functions change as they migrate into supporting a digital collection.
- ✓ Since the collection is digital and there is a worldwide communications infrastructure available, the library no longer must own a copy of information as long as it can provide access.
- ✓ The existing quantity of hardcopy material guarantees that we will not have all digital libraries for at least another generation of technology improvements.
- ✓ But there is no question that libraries have started and will continue to expand their focus to digital formats.
- ✓ With direct electronic access available to users the social aspects of congregating in a library and learning from librarians, friends and colleagues will be lost and new electronic collaboration equivalencies will come into existence (Wiederhold-95).
- ✓ Indexing is one of the critical disciplines in library science and significant effort has gone into the establishment of indexing and cataloging standards.
- ✓ Migration of many of the library products to a digital format introduces both opportunities and challenges.
- ✓ The conversion of existing hardcopy text, images (e.g., pictures, maps) and analog (e.g., audio, video) data and the storage and retrieval of the digital version is a major concern to Digital Libraries which is not considered in information systems.
- ✓ Other issues such as how to continue to provide access to digital information over many years as digital formats change have to be answered for the long term viability of digital libraries.

DATA WAREHOUSES

- ✓ The term Data Warehouse comes more from the commercial sector than academic sources.
- ✓ It comes from the need for organizations to control the proliferation of digital information ensuring that it is known and recoverable.
- ✓ Its goal is to provide to the decision makers the critical information to answer future direction questions.
- ✓ Frequently a data warehouse is focused solely on structured databases.
- ✓ A data warehouse consists of the data, an information directory that describes the contents and meaning of the data being stored, an input function that captures data and moves it to the data warehouse, data search and manipulation tools that allow users the means to access and analyze the warehouse data and a delivery mechanism to export data to other warehouses, data marts (small warehouses or subsets of a larger warehouse), and external systems.
- ✓ Data warehouses are similar to information storage and retrieval systems in that they both have a need for search and retrieval of information.
- ✓ But a data warehouse is more focused on structured data and decision support technologies.
- ✓ In addition to the normal search process, a complete system provides a flexible set of analytical tools to "mine" the data. Data mining (originally called Knowledge Discovery in Databases - KDD) is a search process that automatically analyzes data and extract relationships and dependencies that were not part of the database design.
- ✓ Most of the research focus is on the statistics, pattern recognition and artificial intelligence algorithms to detect the hidden relationships of data.
- ✓ In reality the most difficult task is in preprocessing the data from the database for processing by the algorithms.

- ✓ This differs from clustering in information retrieval in that clustering is based upon known characteristics of items, whereas data mining does not depend upon known relationships.

INFORMATION RETRIEVAL SYSTEM CAPABILITIES

Major functions that are available in an Information Retrieval System are

1. Searching Capabilities.
2. Browsing Capabilities.
3. Miscellenious Capabilities.

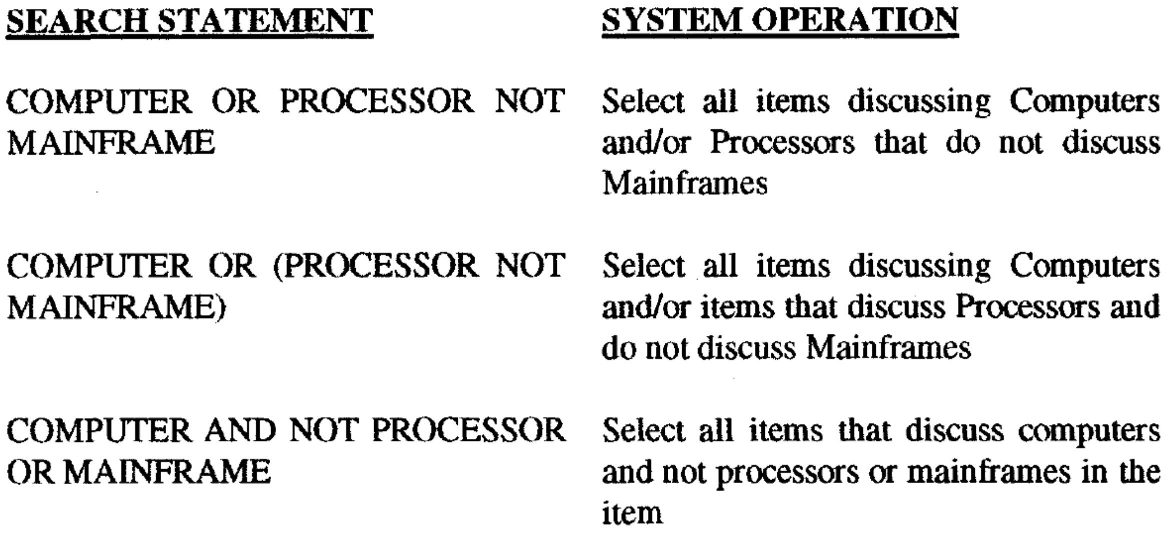
SEARCH CAPABILITIES

- ✓ The objective of the search capability is to allow for a mapping between a user's specified need and the items in the information database that will answer that need.
- ✓ The search query statement is the means that the user employs to communicate a description of the needed information to the system.
- ✓ It can consist of natural language text in composition style and/or query terms (referred to as terms in this book) with Boolean logic indicators between them.
- ✓ One concept that has occasionally been implemented in commercial systems (e.g., RetrievalWare), and holds significant potential for assisting in the location and ranking of relevant items, is the "weighting" of search terms.
- ✓ This would allow a user to indicate the importance of search terms in either a Boolean or natural language interface.

Boolean Logic

- ✓ Boolean logic allows a user to logically relate multiple concepts together to define what information is needed.

- ✓ Typically the Boolean functions apply to processing tokens identified anywhere within an item.
- ✓ The typical Boolean operators are AND, OR, and NOT.
- ✓ These operations are implemented using set intersection, set union and set difference procedures.
- ✓ In the examples of effects of Boolean operators given in Figure 2.1, no precedence order is given to the operators and queries are processed Left to Right unless parentheses are included.



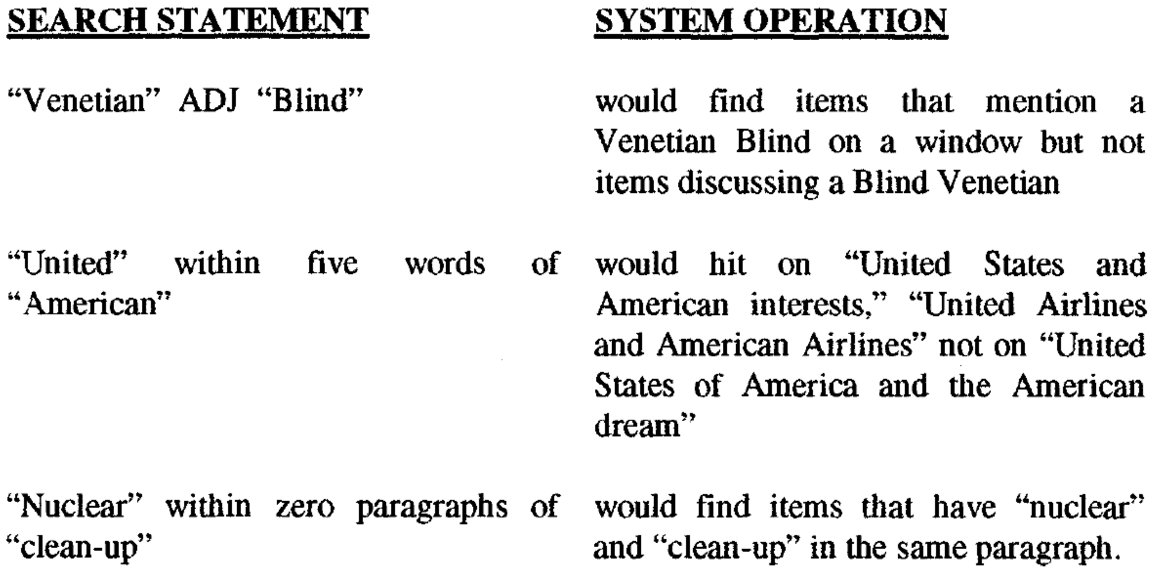
SEARCH STATEMENT	SYSTEM OPERATION
COMPUTER OR PROCESSOR NOT MAINFRAME	Select all items discussing Computers and/or Processors that do not discuss Mainframes
COMPUTER OR (PROCESSOR NOT MAINFRAME)	Select all items discussing Computers and/or items that discuss Processors and do not discuss Mainframes
COMPUTER AND NOT PROCESSOR OR MAINFRAME	Select all items that discuss computers and not processors or mainframes in the item

Figure 2.1 Use of Boolean Operators

Proximity

- ✓ Proximity is used to restrict the distance allowed within an item between two search terms.
- ✓ The semantic concept is that the closer two terms are found in a text the more likely they are related in the description of a particular concept.
- ✓ Proximity is used to increase the precision of a search.
- ✓ If the terms COMPUTER and DESIGN are found within a few words of each other then the item is more likely to be discussing the design of computers than if the words are paragraphs apart.

- ✓ Some proximity search statement examples and their meanings are given in Figure 2.2.



SEARCH STATEMENT	SYSTEM OPERATION
“Venetian” ADJ “Blind”	would find items that mention a Venetian Blind on a window but not items discussing a Blind Venetian
“United” within five words of “American”	would hit on “United States and American interests,” “United Airlines and American Airlines” not on “United States of America and the American dream”
“Nuclear” within zero paragraphs of “clean-up”	would find items that have “nuclear” and “clean-up” in the same paragraph.

Figure 2.2 Use of Proximity

Continuous Word Phrases (CWP)

- ✓ A Contiguous Word Phrase (CWP) is both a way of specifying a query term and a special search operator.
- ✓ A Contiguous Word Phrase is two or more words that are treated as a single semantic unit.
- ✓ An example of a CWP is "United States of America." It is four words that specify a search term representing a single specific semantic concept (a country) that can be used with any of the operators discussed above.
- ✓ Thus a query could specify "manufacturing" AND "United States of America" which returns any item that contains the word "manufacturing" and the contiguous words "United States of America."

Fuzzy Searches

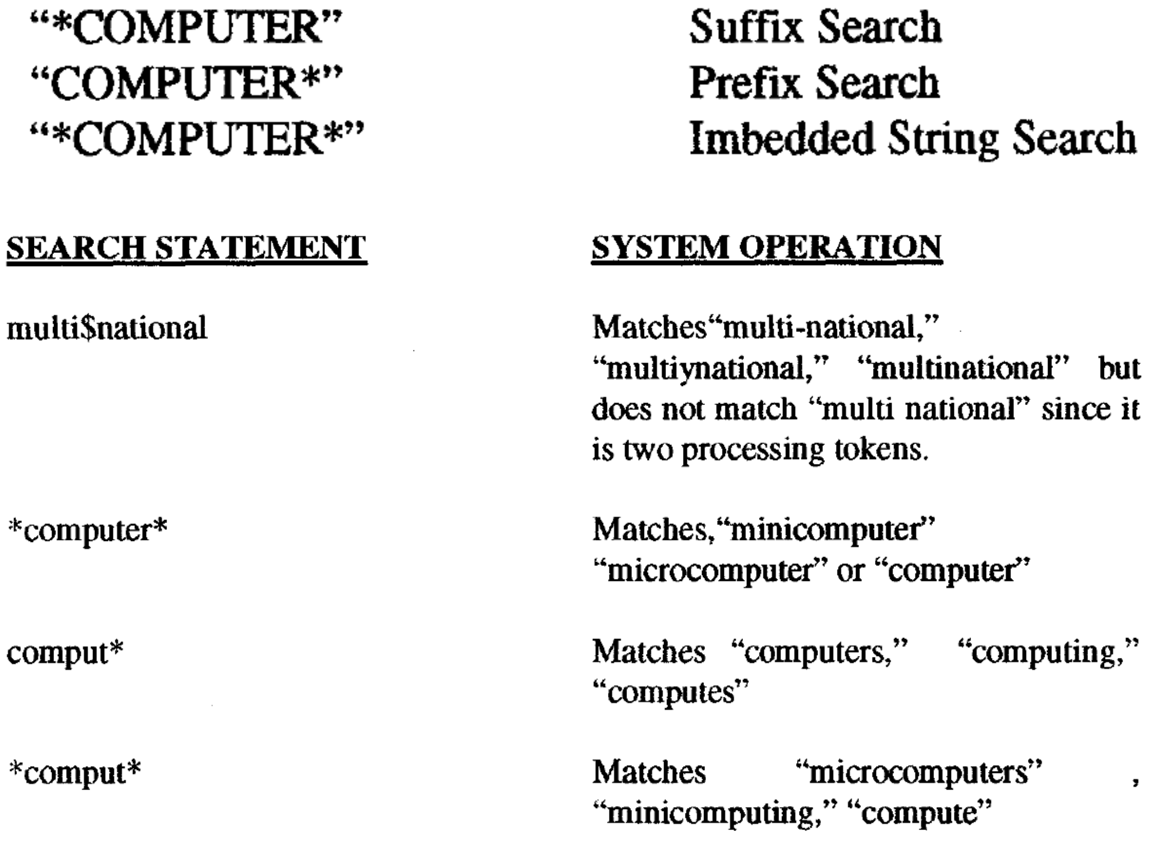
- ✓ Fuzzy Searches provide the capability to locate spellings of words that are similar to the entered search term. This function is primarily used to compensate for errors in spelling of words.
- ✓ Fuzzy searching increases recall at the expense of decreasing precision (i.e., it can erroneously identify terms as the search term).
- ✓ In the process of expanding a query term fuzzy searching includes other terms that have similar spellings, giving more weight (in systems that rank output to words in the database that have similar word lengths and position of the characters as the entered term.
- ✓ A Fuzzy Search on the term "computer" would automatically include the following words from the information database: "computer," "compiter," "conputer," "computter," "compute."

Term Masking

- ✓ Term masking is the ability to expand a query term by masking a portion of the term and accepting as valid any processing token that maps to the unmasked portion of the term.
- ✓ The value of term masking is much higher in systems that do not perform stemming or only provide a very simple stemming algorithm.
- ✓ There are two types of search term masking: fixed length and variable length. Sometimes they are called fixed and variable length "don't care" functions.
- ✓ Fixed length masking is a single position mask. It masks out any symbol in a particular position or the lack of that position in a word. Figure 2.3 gives an example of fixed term masking.
- ✓ Variable length "don't cares" allows masking of any number of characters within a processing token. The masking may be in the front, at the end, at both front and end, or imbedded. The first three of these cases are called

suffix search, prefix search and imbedded character string search, respectively.

- ✓ The use of an imbedded variable length don't care is seldom used. Figure 2.3 provides examples of the use of variable length term masking.
- ✓ If "*" represents a variable length don't care then the following are examples of its use:



"*COMPUTER"	Suffix Search
"COMPUTER*"	Prefix Search
"*COMPUTER*"	Imbedded String Search

SEARCH STATEMENT	SYSTEM OPERATION
multi$national	Matches"multi-national," "multiynational," "multinational" but does not match "multi national" since it is two processing tokens.
computer	Matches,"minicomputer" "microcomputer" or "computer"
comput*	Matches "computers," "computing," "computes"
comput	Matches "microcomputers" , "minicomputing," "compute"

Figure 2.3 Term Masking

Numeric and Date Ranges

- ✓ Term masking is useful when applied to words, but does not work for finding ranges of numbers or numeric dates.
- ✓ To find numbers larger than "125," using a term "125"" will not find any number except those that begin with the digits "125."

- ✓ Systems, as part of their normalization process, characterizes words as numbers or dates. This allows for specialized numeric or date range processing against those words.

Concept/Thesaurus Expansion

- ✓ Associated with both Boolean and Natural Language Queries is the ability to expand the search terms via Thesaurus or Concept Class database reference tool.
- ✓ A Thesaurus is typically a one-level or two-level expansion of a term to other terms that are similar in meaning.
- ✓ A Concept Class is a tree structure that expands each meaning of a word into potential concepts that are related to the initial term (e.g., in the TOPIC system).
- ✓ Concept classes are sometimes implemented as a network structure that links word stems (e.g., in the Retrieval Ware system).
- ✓ An example of Thesaurus and Concept Class structures are shown in Figure 2.4 (Thesaurus-93) and Figure 2.5.

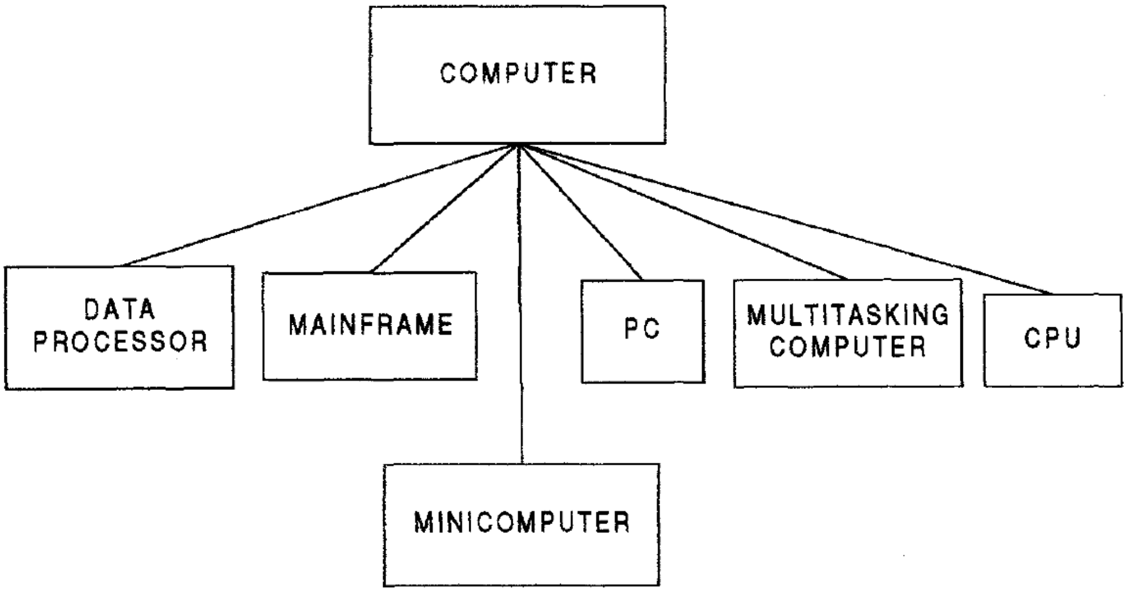
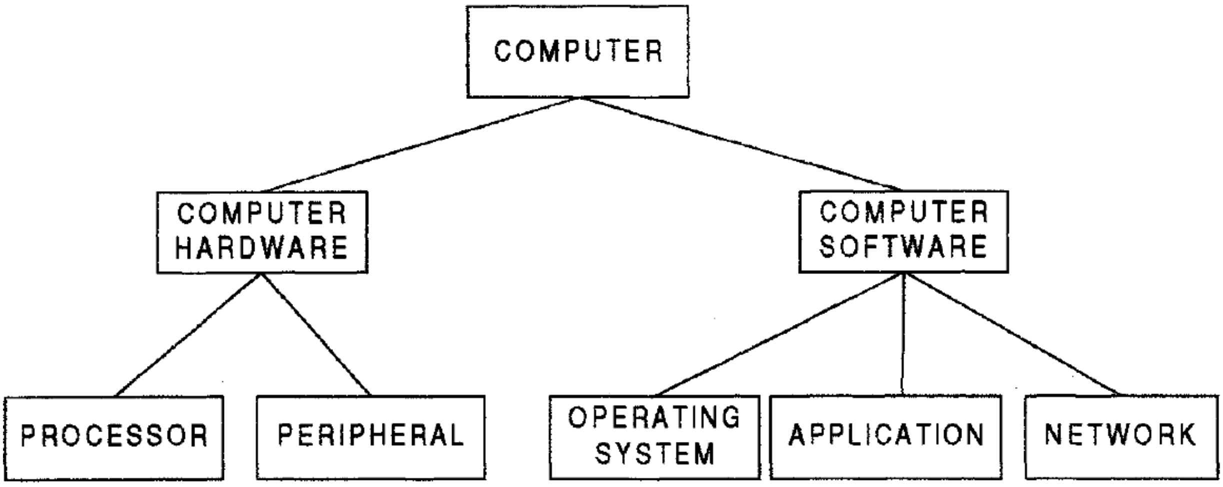


Figure 2.4 Thesaurus for term "computer"



Natural Language Queries

- ✓ Rather than having the user enter a specific Boolean query by specifying search terms and the logic between them, Natural Language Queries allow a user to enter a prose statement that describes the information that the user wants to find.
- ✓ An example of a Natural Language Query is: Find for me all the items that discuss oil reserves and current attempts to find new oil reserves. Include ally items that discuss the international financial aspects of the oil production process. Do not include items about the oil industry in the United States.

BROWSE CAPABILITIES

- ✓ Once the search is complete, Browse capabilities provide the user with the capability to determine which items are of interest and select those to be displayed.
- ✓ There are two ways of displaying a summary of the items that are associated with a query: line item status and data visualization.
- ✓ From these summary displays, the user can select the specific items mid zones within the items for display.

- ✓ The system also allows for easy transitioning between die summary displays and review of specific items.
- ✓ If searches resulted in high precision, then the importance of the browse capabilities would be lessened.
- ✓ Since searches return many items that are not relevant to the user's information need, browse capabilities can assist the user in focusing on items that have the highest likelihood in meeting his need.
- ✓ There are two ways of displaying a summary of the items that are associated with a query: Line Item Status and Data Visualization.

Different browsing capabilities are

1. Ranking

- ✓ Under Boolean systems, the status display is a count of the number of items found by the query.
- ✓ Every one of the items meet all aspects of the Boolean query.
- ✓ The reasons why an item was selected can easily be traced to and displayed (e.g., via highlighting) in the retrieved items.
- ✓ Hits are retrieved in either a sorted order (e.g., sort by Title) or in time order from the newest to the oldest item.
- ✓ With the introduction of ranking based upon predicted relevance values, the status summary displays the relevance score associated with the item along with a brief descriptor of the item (usually both fit on one display screen line).
- ✓ The relevance score is an estimate of the search system on how closely the item satisfies the search statement.
- ✓ Typically relevance scores are normalized to a value between 0.0 and 1.0.
- ✓ The highest value of 1.0 is interpreted that the system is sure that the item is relevant to the search statement.

- ✓ This allows the user to determine at what point to stop reviewing items because of reduced likelihood of relevance.
- ✓ Theoretically every item in the system could be returned but many of the items will have a relevance value of 0.0 (not relevant).
- ✓ Practically, systems have a default minimum value which the user can modify that stops returning items that have a relevance value below the specified value.

2. Zoning

- ✓ When the user displays a particular item, the objective of minimization of overhead still applies.
- ✓ The user wants to see the minimum information needed to determine if the item is relevant.
- ✓ Once the determination is made an item is possibly relevant, the user wants to display the complete item for detailed review.
- ✓ Limited display screen sizes require selectability of what portions of an item a user needs to see to make the relevance determination.

3. Highlighting

- ✓ Another display aid is an indication of why an item was selected. This indication, frequently highlighting, lets the user quickly focus on the potentially relevant parts of the text to scan for item relevance.
- ✓ Different strengths of highlighting indicates how strongly the highlighted word participated in the selection of the item.
- ✓ Most systems allow the display of an item to begin with the first highlight within tile item and allow subsequent jumping to the next highlight.
- ✓ Another capability, which is gaining strong acceptance, is for the system to determine the passage in the document most relevant to the query and position the browse to start at that passage.

- ✓ Highlighting has always been useful in Boolean systems to indicate the cause of the retrieval. This is because of the direct mapping between the terms in the search and the terms in the item.

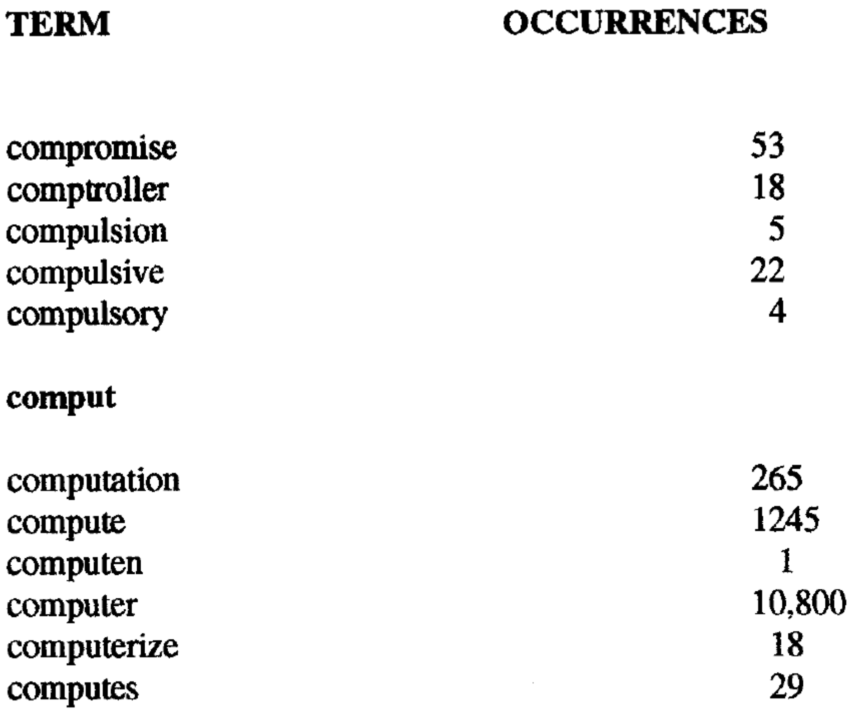
MISCELLANEOUS CAPABILITIES

- ✓ There are many additional functions that facilitate the user's ability to input queries, reducing the time it takes to generate the queries, and reducing a priori the probability of entering a poor query.
- ✓ Vocabulary browse provides knowledge on the processing tokens available in the searchable database and their distribution in terms of items within the database.
- ✓ Iterative searching and search history logs summarize previous search activities by the user allowing access to previous results from the current user session.
- ✓ Canned queries allow access to queries generated and saved in previous user sessions.

Vocabulary Browse

- ✓ Vocabulary Browse provides the capability to display in alphabetical sorted order words from the document database.
- ✓ Logically, all unique words (processing tokens) in the database are kept in sorted order along with a count of the number of unique items in which the word is found.
- ✓ The user can enter a word or word fragment and the system will begin to display file dictionary around the entered text.
- ✓ Figure 2.6 shows what is seen in vocabulary browse if the user enters "comput."

- ✓ The system indicates what word fragement the user entered and then alphabetically displays other words found in the database in collating sequence on either side of file entered term.
- ✓ The user can continue scrolling in either direction reviewing additional terms in the database.
- ✓ Vocabulary browse provides information on the exact words in the database.
- ✓ It helps the user determine the impact of using a fixed or variable length mask on a search term and potential mis-spellings.
- ✓ The user can determine that entering the search term "compul*" in effect is searching for "compulsion" or "compulsive" or "compulsory." It also shows that someone probably entered the word "computen" when they really meant "computer.



TERM	OCCURRENCES
compromise	53
comptroller	18
compulsion	5
compulsive	22
compulsory	4
comput	
computation	265
compute	1245
computen	1
computer	10,800
computerize	18
computes	29

Figure 2.6 Vocabulary Browse List with entered term “comput”

lterative Search and Search History Log

- ✓ Frequently a search returns a Hit file containing many more items than the user wants to review.
- ✓ Rather than typing in a complete new query, the results of the previous search can be used as a constraining list to create a new query that is applied against it.
- ✓ This has the same effect as taking the original query and adding additional search statement against it in an AND condition.
- ✓ This process of refining the results of a previous search to focus on relevant items is called iterative search.

Canned Query

The capability to name a query and store it to be retrieved and executed during a later user session is called canned or stored queries.

Multimedia

Once a list of potential items that Satisfy the query are discovered, the techniques for displaying them when they are multimedia Introduces new Challenges.

******

UNIT – II

CONTENTS

1. **Cataloging and Indexing:** History and Objectives of Indexing, Indexing Process, Automatic Indexing, Information Extraction.
2. **Data Structure:** Introduction to Data Structure, Stemming Algorithms, Inverted File Structure, N-Gram Data Structures, PAT Data Structure, Signature File Structure, Hypertext and XML Data Structures, Hidden Markov Models.

CATALOGING AND INDEXING

HISTORY AND OBJECTIVES OF INDEXING

Indexing

The transformation from received item to searchable data structure is called indexing. Process can be manual or automatic.

History

- ✓ Indexing (originally called Cataloging) is the oldest technique for identifying the contents of items to assist in their retrieval.
- ✓ The objective of cataloging is to give access points to a collection that are expected and most useful to the users of the information.
- ✓ The basic information required on an item, what is the item and what it is about, has not changed over the centuries.
- ✓ As early as the third-millennium, in Babylon, libraries of cuneiform tablets were arranged by subject (Hyman-89).
- ✓ Up to the 19th Century there was little advancement in cataloging, only changes in the methods used to represent the basic information (Norris-69).
- ✓ In the late 1800s subject indexing became hierarchical (e.g., Dewey Decimal System).

- ✓ In 1963 the Library of Congress initiated a study on the computerization of bibliographic surrogates.
- ✓ From 1966 - 1968 the Library of Congress ran its MARC I pilot project.
- ✓ MARC (MAchine Readable Cataloging) standardizes the structure, contents and coding of bibliographic records.
- ✓ The system became operational in 1969 (Avram-75).
- ✓ The earliest commercial cataloging system is DIALOG, which was developed by Lockheed Corporation in 1965 for NASA.
- ✓ It became commercial in 1978 with three government files of indexes to technical publications.
- ✓ By 1988, when it was sold to Knight-Ridder, DIALOG contained over 320 index databases used by over 91,000 subscribers in 86 countries (Harper-81).
- ✓ In the 1990s, the significant reduction in cost of processing power and memory in modern computers, along with access to the full text of an item from the publishing stages in electronic form, allow use of the full text of an item as an alternative to the indexer-generated subject index.

Objectives

1. The public file indexer needs to consider the information needs of all users of library system. Items overlap between full item indexing, public and private indexing of files.
2. Users may use public index files as part of search criteria to increase recall.
3. They can constrain there search by private index files.
4. The primary objective of representing the concepts within an item to facilitate users finding relevant information.
5. Users may use public index files as part of search criteria to increase recall.
6. They can constrain there search by private index files.

7. The primary objective of representing the concepts within an item to facilitate users finding relevant information.

INDEXING PROCESS

- ✓ When an organization with multiple indexers decides to create a public or private index some procedural decisions on how to create the index terms assist the indexers and end users in knowing what to expect in the index file.
- ✓ The first decision is the scope of the indexing to define what level of detail the subject index will contain. This is based upon usage scenarios of the end users.
- ✓ The other decision is the need to link index terms together in a single index for a particular concept.
- ✓ Linking index terms is needed when there are multiple independent concepts found within an item.

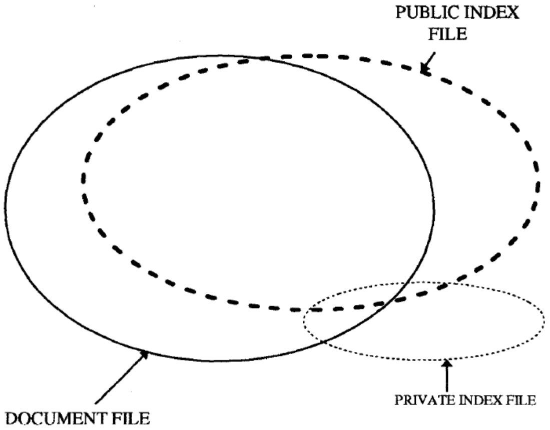


Figure 3.1 Items Overlap Between Full Item Indexing, Public File Indexing and Private File Indexing

Scope of Indexing

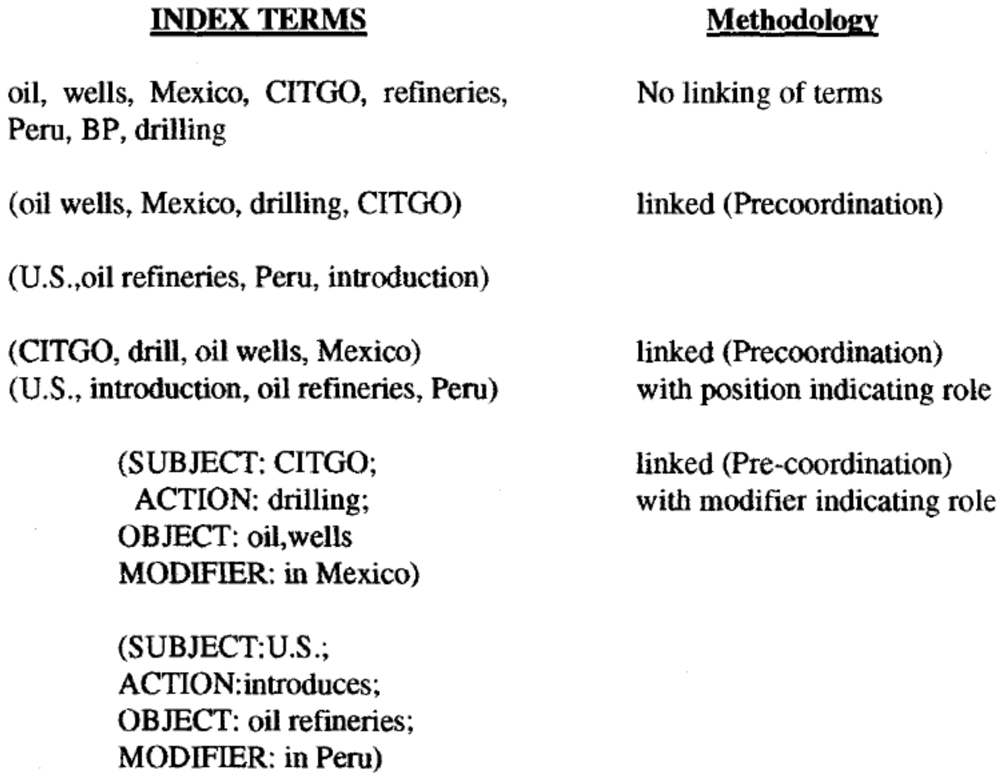
- ✓ When performed manually, the process of reliably and consistently determining the bibliographic terms that represent the concepts in an item is extremely difficult.
- ✓ Problems arise from interaction of two sources: the author and the indexer.

- ✓ The vocabulary domain of the author may be different than that of the indexer, causing the indexer to misinterpret the emphasis and possibly even the concepts being presented.
- ✓ The indexer is not an expert on all areas and has different levels of knowledge in the different areas being presented in the item.
- ✓ This results in different quality levels of indexing.
- ✓ The indexer must determine when to stop the indexing process.
- ✓ There are two factors involved in deciding on what level to index tile concepts in an item: the exhaustivity and the specificity of indexing desired.
- ✓ Exhanstivity of indexing is the extent to which the different concepts in the item are indexed. For example, if two sentences of a 10-page item on microprocessors discusses on-board caches, should this concept be indexed?
- ✓ Specificity relates to the preciseness of the index terms used in indexing. For example, whether the term "processor" or the term "microcomputer" or the term "Pentium" should be used in the index of an item is based upon the specificity decision.
- ✓ Indexing an item only on the most important concept in it and using general index terms yields low exhaustivity and specificity.
- ✓ This approach requires a minimal number of index terms per item and reduces the cost of generating the index.

Precoordination and Linkages

- ✓ Another decision on the indexing process is whether linkages are available between index terms for an item.
- ✓ Linkages are used to correlate related attributes associated with concepts discussed in an item.

- ✓ This process of creating term linkages at index creation time is called pre-coordination.
- ✓ When index terms are not coordinated at index time, the coordination occurs at search time. This is called post-coordination.
- ✓ Figure 3.2 shows the different types of linkages.



INDEX TERMS	Methodology
oil, wells, Mexico, CITGO, refineries, Peru, BP, drilling	No linking of terms
(oil wells, Mexico, drilling, CITGO)	linked (Precoordination)
(U.S.,oil refineries, Peru, introduction)	
(CITGO, drill, oil wells, Mexico) (U.S., introduction, oil refineries, Peru)	linked (Precoordination) with position indicating role

(SUBJECT: CITGO;
ACTION: drilling;
OBJECT: oil,wells
MODIFIER: in Mexico)
	linked (Pre-coordination)
	with modifier indicating role

(SUBJECT:U.S.;
ACTION:introduces;
OBJECT: oil refineries;
MODIFIER: in Peru)	

Figure 3.2 Linkage of Index Terms

AUTOMATIC INDEXING

- ✓ Automatic indexing is the capability for the system to automatically determine the index terms to be assigned to an item.
- ✓ The simplest case is when all words in the document are used as possible index terms (total document indexing).
- ✓ More complex processing is required when the objective is to emulate a human indexer and determine a limited number of index terms for the major concepts in the item.

- ✓ As discussed, the advantages of human indexing are the ability to determine concept abstraction and judge the value of a concept.
- ✓ The disadvantages of human indexing over automatic indexing are cost, processing time and consistency.
- ✓ Once the initial hardware cost is amortized, the costs of automatic indexing are absorbed as part of the normal operations and maintenance costs of the computer system.
- ✓ There are no additional indexing costs versus the salaries and benefits regularly paid to haman indexers.
- ✓ Processing time of an item by a human indexer varies significantly based upon the indexer's knowledge of the concepts being indexed, the exhaustivity and specificity guidelines and the amount and accuracy of preprocessing via Automatic File Build.
- ✓ Even for relatively short items (e.g., 300 - 500 words) it normally takes at least five minutes per item.
- ✓ A significant portion of this time is caused by the human interaction with the computer (e.g., typing speeds, cursor positioning, correcting spelling errors, taking breaks between activities).
- ✓ Automatic indexing requires only a few seconds or less of computer time based upon the size of the processor and the complexity of the algorithms to generate the index.
- ✓ Another advantage to automatic indexing is the predictably of algorithms.

Indexing by Term

- ✓ When the terms of the original item are used as a basis of the index process, there are two major techniques for creation of the index: statistical and natural language.
- ✓ Statistical techniques can be based upon vector models and probabilistic models with a special case being Bayesian models. They are classified as

statistical because their calculation of weights use statistical information such as the frequency of occurrence of words and their distributions in the searchable database.

- ✓ Natural language techniques also use some statistical information, but perform more complex parsing to define the final set of index concepts.
- ✓ Figure 3.3 shows the basic weighting approach for index terms or associations between query terms and index terms.

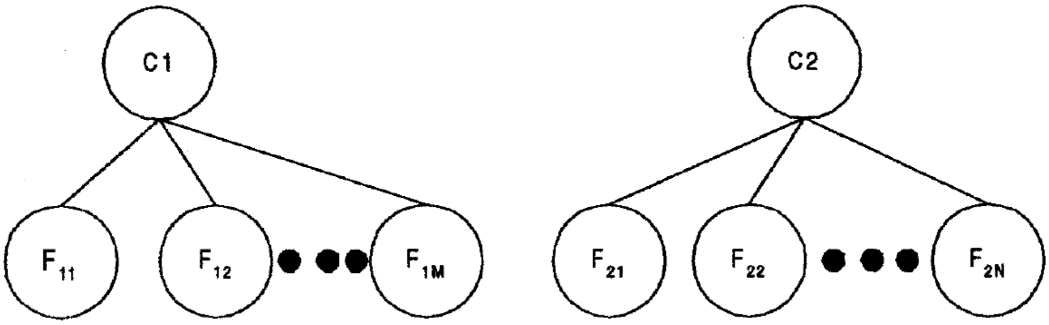
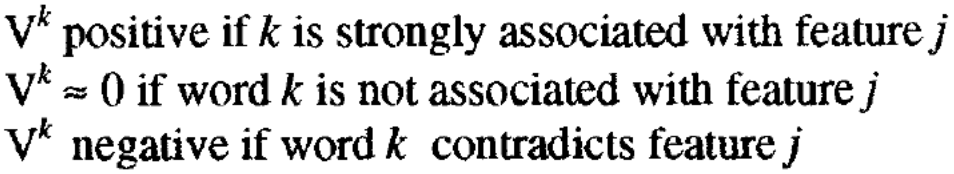


Figure 3.3 Two-level Bayesian network

Indexing by Concept

- ✓ The basis for concept indexing is that there are many ways to express the same idea and increased retrieval performance comes from using a single representation.
- ✓ Indexing by term treats each of these occurrences as a different index and then uses thesauri or other query expansion techniques to expand a query to find the different ways the same thing has been represented.
- ✓ Concept indexing determines a canonical set of concepts based upon a test set of terms and uses them as a basis for indexing all items. This is also called Latent Semantic Indexing because it is indexing the latent semantic information in items.
- ✓ For any word stem k, its context vector V k is an n-dimensional vector with each component j interpreted as follows:



V^k positive if k is strongly associated with feature j

$V^k \approx 0$ if word k is not associated with feature j

V^k negative if word k contradicts feature j

INFORMATION EXTRACTION

- ✓ There are two processes associated with information extraction:
  1. Determination of facts to go into structured fields in a database.
  2. Extraction of text that can be used to summarize an item.
- ✓ The process of extracting facts to go into indexes is called Automatic File Build.
- ✓ In establishing metrics to compare information extraction, precision and recall are applied with slight modifications.
- ✓ Recall refers to how much information was extracted from an item versus how much should have been extracted from the item.
- ✓ It shows the amount of correct and relevant data extracted versus the correct and relevant data in the item.
- ✓ Precision refers to how much information was extracted accurately versus the total information extracted.
- ✓ Additional metrics used are over generation and fallout.
- ✓ Over generation measures the amount of irrelevant information that is extracted.
- ✓ This could be caused by templates filled on topics that are not intended to be extracted or slots that get filled with non-relevant data.
- ✓ Fallout measures how much a system assigns incorrect slot fillers as the number of.
- ✓ These measures are applicable to both human and automated extraction processes.
- ✓ Another related information technology is document summarization.

- ✓ Rather than trying to determine specific facts, the goal of document summarization is to extract a summary of an item maintaining the most important ideas while significantly reducing the size.
- ✓ Examples of summaries that are often part of any item are titles, table of contents, and abstracts with the abstract being the closest.
- ✓ The abstract can be used to represent the item for search purposes or as a way for a user to determine the utility of an item without having to read the complete item.

DATA STRUCTURE

INTRODUCTION TO DATA STRUCTURE

- ✓ The knowledge of data structure gives an insight into the capabilities available to the system.
- ✓ There are usually two major data structures in any information system.
- ✓ One structure stores and manages the received items in their normalized form.
- ✓ The process supporting this structure is called the "document manager."
- ✓ The other major data structure contains the processing tokens and associated data to support search.
- ✓ Figure 4.1 expands file document file creation function.
- ✓ It does not address the document management function nor the data structures and other related theory associated with the parsing of queries.
- ✓ For that background the reader should pursue a text on finite automata and language (regular expressions).

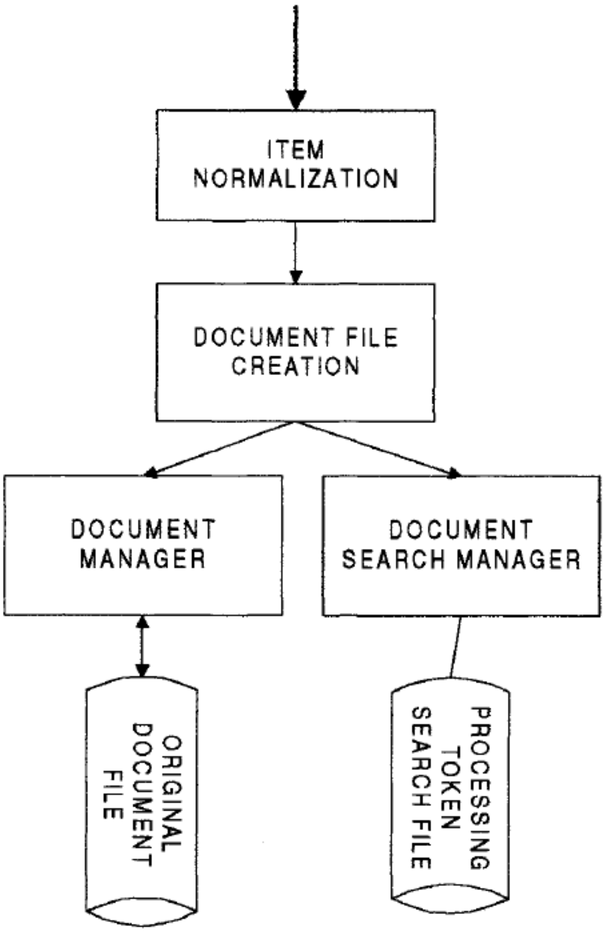


Figure 4.1 Major Data Structures

- ✓ A special data structure that is becoming common place because of its use on the Internet is hypertext.
- ✓ This structure allows the creator of an item to manually or automatically create imbedded links within one item to a related item.

STEMMING ALGORITHMS

- ✓ Tile concept of stemming has been applied to information systems from theft initial automation in the 1960's.
- ✓ The original goal of stemming was to improve performance and require less system resources by reducing the number of unique words that a system has to contain.
- ✓ With tile continued significant increase in storage and computing power, use of stemming for performance reasons is no longer as important.

- ✓ Stemming is now being reviewed for tile potential improvements it can make in recall versus its associated decline in precision.
- ✓ A system designer can trade off the increased overhead of stemming in creating processing tokens versus reduced search time overhead of processing query terms with trailing "don't cares" to include all of their variants.
- ✓ The stemming process creates one large index for the stem versus Term Masking which requires the merging (ORing) of the indexes for every term that matches the search term.

Introduction to the Stemming Process

- ✓ Stemming algorithms are used to improve the efficiency of the information system and to improve recall.
- ✓ Conflation is the term frequently used to refer to mapping multiple morphological variants to a single representation (stem).
- ✓ The premise is that the stem carries the meaning of the concept associated with the word and the that affixes (endings) introduce subtle modifications to the concept or are used for syntactical purposes.
- ✓ Languages have precise grammars that define their usage, but also evolve based upon human usage.
- ✓ Thus exceptions and non-consistent variants are always present in languages that typically require exception look-up tables in addition to the normal reduction rules.
- ✓ At first glance, the idea of equating multiple representations of a word as a single stem term would appear to provide significant compression, with associated savings in storage and processing.
- ✓ For example, the stem "comput" could associate "computable, computability, computation, computational, computed, computing, computer, computerese, computerize" to one compressed word.

- Another major use of stemming is to improve recall.
- As long as a semantically consistent stem can be identified for a set of words, the generalization process of stemming does help in not missing potentially relevant items.
- Stemming of the words "calculate, calculates, calculation, calculations, calculating" to a single stem ("calculat") insures whichever of those terms is entered by the user, it is translated to the stem and finds all the variants in any items they exist.
- In contrast, stemming can not improve, but has the potential for decreasing precision.
- Stemming can also cause problems for Natural Language Processing (NLP) systems by causing the loss of information needed for aggregate levels of natural language processing (discourse analysis).

Porter Stemming Algorithm

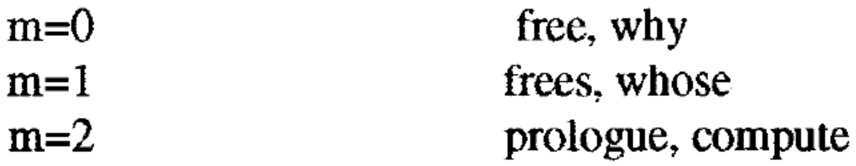
The Porter Algorithm is based upon a set of conditions of the stem, suffix and prefix and associated actions given the condition. Some examples of stem conditions are:

1. The measure, m, of a stem is a function of sequences of vowels (a, e, i, o, u, y) followed by a consonant. If V is a sequence of vowels and C is a sequence of consonants, then m is:

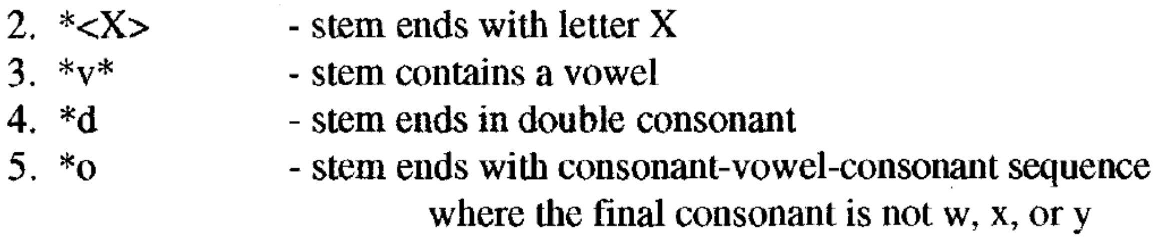
$$C(VC)^{m}V$$

where the initial C and final V are optional and m is the number VC repeats.

Measure	Example

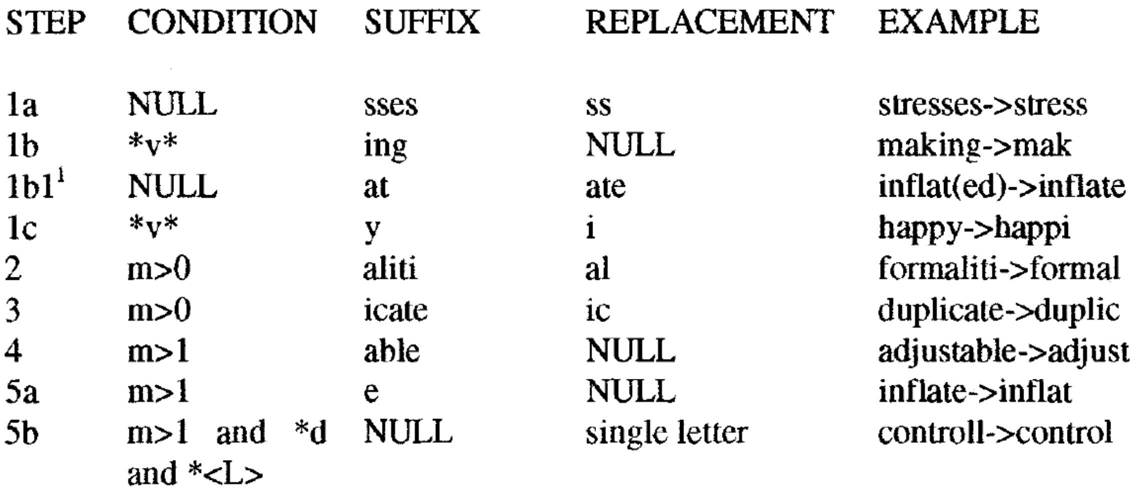


m=0	free, why
m=1	frees, whose
m=2	prologue, compute



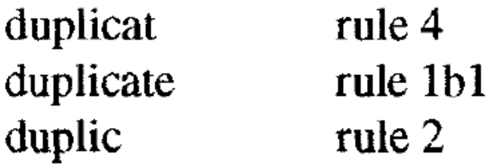
2. *<X> - stem ends with letter X
3. *v* - stem contains a vowel
4. *d - stem ends in double consonant
5. *o - stem ends with consonant-vowel-consonant sequence where the final consonant is not w, x, or y

Rules are divided into steps to define the order of applying the rules. The following are some examples of the rules:



STEP	CONDITION	SUFFIX	REPLACEMENT	EXAMPLE
1a	NULL	sses	ss	stresses->stress
1b	*v*	ing	NULL	making->mak
1b1[1]	NULL	at	ate	inflat(ed)->inflate
1c	*v*	y	i	happy->happi
2	m>0	aliti	al	formaliti->formal
3	m>0	icate	ic	duplicate->duplic
4	m>1	able	NULL	adjustable->adjust
5a	m>1	e	NULL	inflate->inflat
5b	m>1 and *d and *<L>	NULL	single letter	controll->control

Given the word "duplicatable," the following are the steps in the stemming process:



duplicat	rule 4
duplicate	rule 1b1
duplic	rule 2

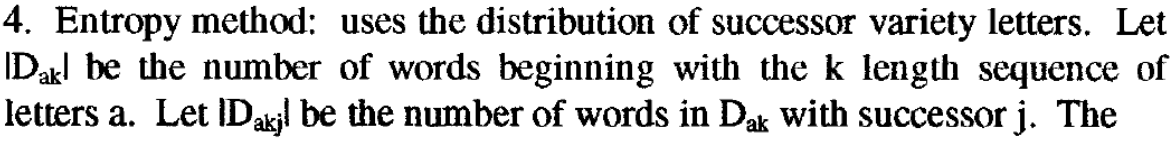
Dictionary Look-Up Stemmers

- ✓ An alternative to solely relying on algorithms to determine a stem is to use a dictionary look-up mechanism.

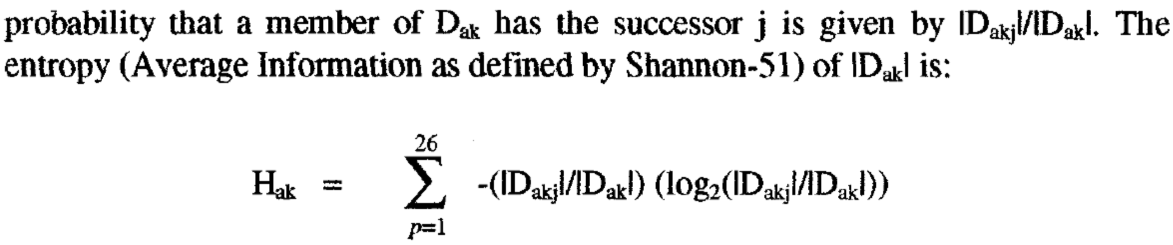
- ✓ In this approach, simple stemming rules still may be applied. The rules are taken from those that have the fewest exceptions (e.g., removing pluralization from nouns).
- ✓ But even the most consistent rules have exceptions that need to be addressed.
- ✓ The original term or stemmed version of the term is looked up in a dictionary and replaced by the stem that best represents it.
- ✓ This technique has been implemented in the INQUERY and Retrieval Ware Systems.

Successor Stemmers

- ✓ Successor stemmers are based upon the length of prefixes that optimally stem expansions of additional suffixes.
- ✓ The algorithm is based upon an analogy in structural linguistics that investigated word and morpheme boundaries based upon file distribution of phonemes, the smallest unit of speech that distinguish one word from another (Hafer-74).
- ✓ The process determines the successor varieties for a word, uses this information to divide a word into segments and selects one of the segments as the stem.
- ✓ The successor varieties of a word are used to segment a word by applying one of the following four methods:
    1. **Cutoff method:** A cutoff value is selected to define stem length. The value varies for each possible set of words.
    2. **Peak and Plateau:** A segment break is made after a character whose successor variety exceeds that of the character immediately preceding it and the character immediately following it.
    3. **Complete word method:** Break on boundaries of complete words.



4. Entropy method: uses the distribution of successor variety letters. Let $|D_{ak}|$ be the number of words beginning with the k length sequence of letters a. Let $|D_{akj}|$ be the number of words in D_{ak} with successor j. The



probability that a member of D_{ak} has the successor j is given by $|D_{akj}|/|D_{ak}|$. The entropy (Average Information as defined by Shannon-51) of $|D_{ak}|$ is:

$$H_{ak} = \sum_{p=1}^{26} -(|D_{akj}|/|D_{ak}|)\ (\log_2(|D_{akj}|/|D_{ak}|))$$

INVERTED FILE STRUCTURE

- ✓ The most common data structure used in both database management and Information Retrieval Systems is the inverted file structure.
- ✓ Inverted file structures are composed of three basic files: the document file, the inversion lists (sometimes called posting files) and the dictionary.
- ✓ The name "inverted file" comes from its underlying methodology of storing an inversion of the documents: inversion of the document from the perspective that, for each word, a list of documents in which the word is found in is stored (the inversion list for that word).
- ✓ Each document in the system is given a unique numerical identifier. It is that identifier that is stored in the inversion list.
- ✓ The way to locate the inversion list for a particular word is via the Dictionary.
- ✓ The Dictionary is typically a sorted list of all unique words (processing tokens) in the system and a pointer to the location of its inversion list (see Figure 4.5).

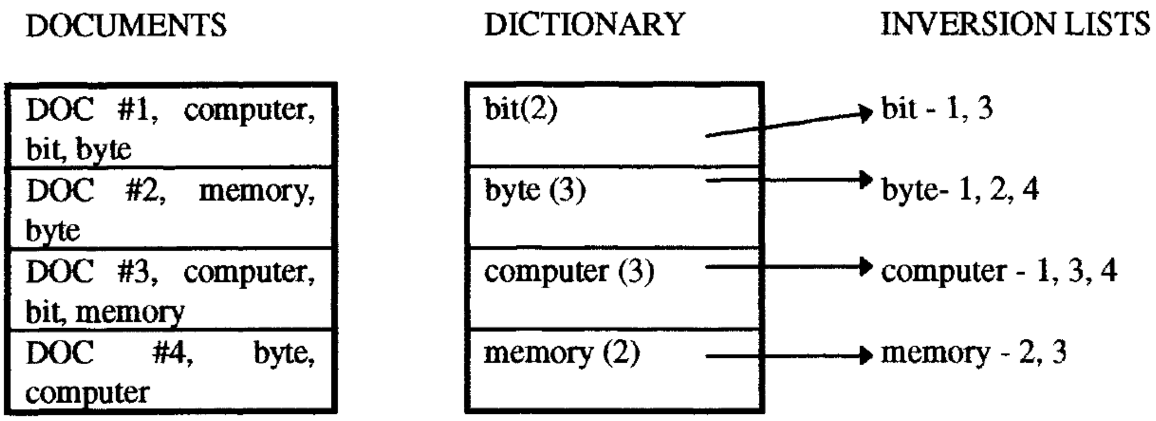


Figure 4.5 Inverted File Structure

- ✓ Rather than using a dictionary to point to the inversion list, B-trees can be used.
- ✓ The inversion lists may be at the leaf level or referenced in higher level pointers.
- ✓ Figure 4.6 shows how the words in Figure 4.5 would appear.
- ✓ A B-tree of order m is defined as:
  1. A root node with between 2 and 2m keys.
  2. All other internal nodes have between m and 2m keys.
  3. All keys are kept in order from smaller to larger.
  4. All leaves are at the same level or differ by at most one level.

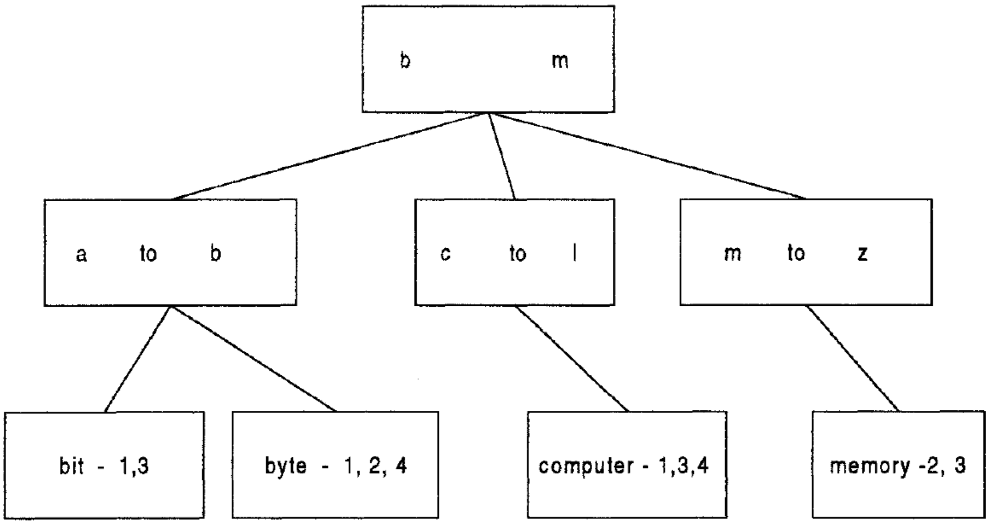


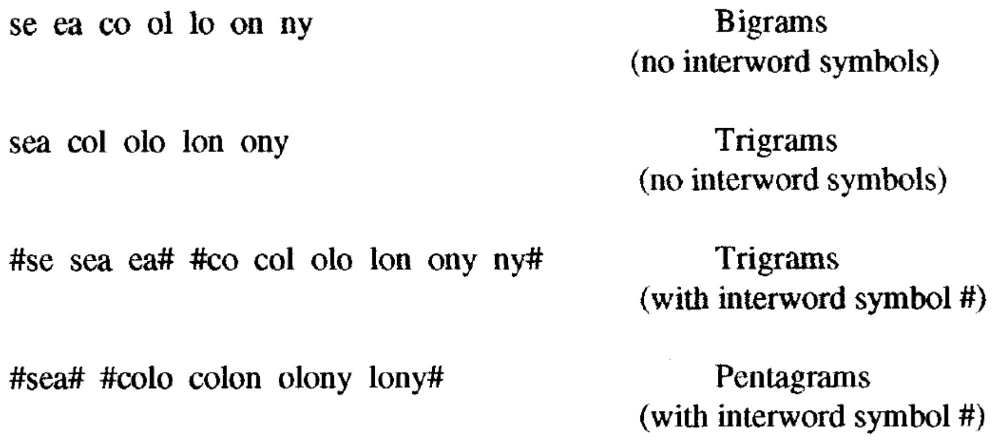
Figure 4-6 B-Tree Inversion Lists

- ✓ Inversion list file structures are well suited to store concepts and their relationships.
- ✓ Each inversion list can be thought of as representing a particular concept.
- ✓ The inversion list is then a concordance of all of the items that contain that concept.
- ✓ Finer resolution of concepts can additionally be maintained by storing locations with an item and weights of the item in the inversion lists.
- ✓ For Natural Language Processing algorithms, other structures may be more appropriate or required in addition to inversion lists for maintaining the required semantic and syntactic information.

N-GRAM DATA STRUCTURES

- ✓ N-Grams can be viewed as a special technique for conflation (stemming) and as a unique data structure in information systems.
- ✓ N-Grams are a fixed length consecutive series of "n" characters.
- ✓ Unlike stemming that generally tries to determine file stem of a word that represents the semantic meaning of the word, ngrams do not care about semantics.
- ✓ Instead they are algorithmically based upon a fixed number of characters.
- ✓ The searchable data structure is transformed into overlapping n-grams, which are then used to create the searchable database.
- ✓ Examples of bigrams, trigrams and pentagrams are given in Figure 4.7 for the word phrase "sea colony."
- ✓ For n-grams, with n greater than two, some systems allow inter word symbols to be part of the n-gram set usually excluding the single character with interword symbol option.
- ✓ The symbol # is used to represent the interword symbol which is anyone of a set of symbols (e.g., blank, period, semicolon, colon, etc.).

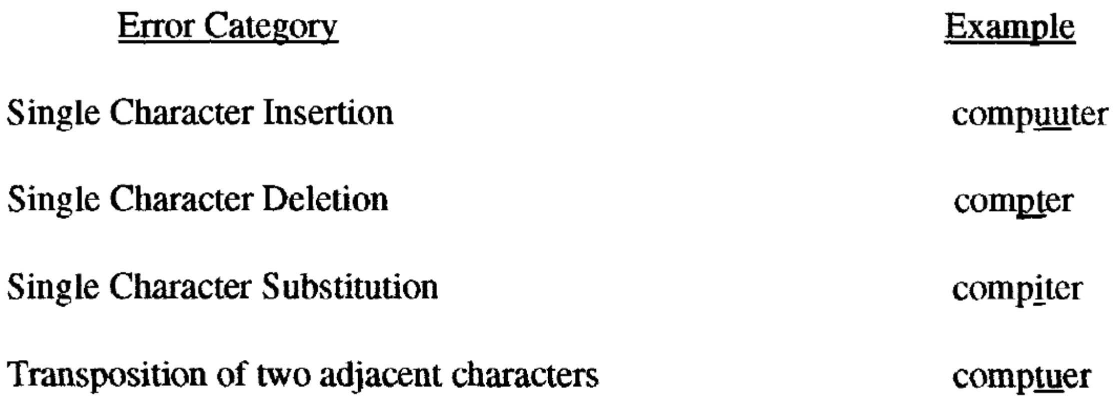
- ✓ Each of the n-grams created becomes a separate processing tokens and are searchable. It is possible that the same n-gram can be created multiple times from a single word.



se ea co ol lo on ny	Bigrams (no interword symbols)
sea col olo lon ony	Trigrams (no interword symbols)
#se sea ea# #co col olo lon ony ny#	Trigrams (with interword symbol #)
#sea# #colo colon olony lony#	Pentagrams (with interword symbol #)

Figure 4.7 Bigrams, Trigrams and Pentagrams for "sea colony"

- ✓ Frequency of occurrence of n gram patterns also can be used for identifying the language of an item (Damashek-95, Cohen-95).



Error Category	Example
Single Character Insertion	compuuter
Single Character Deletion	compter
Single Character Substitution	compiter
Transposition of two adjacent characters	comptuer

Figure 4.8 Categories of Spelling Errors

- ✓ As shown in Figure 4.7, an n-gram is a data structure that ignores words and treats the input as a continuous data, optionally limiting its processing by interword symbols.
- ✓ The data structure consists of fixed length overlapping symbol segments that define the searchable processing tokens.

- ✓ These tokens have logical linkages to all the items in which the tokens are found.
- ✓ Inversion lists, document vectors and other proprietary data structures are used to store the linkage data structure and are used in the search process.
- ✓ In some cases just the least frequently occurring n-gram is kept as part of a first pass search process (Yochum-85).
- ✓ The choice of the fixed length word fragment size has been studied in many contexts.
- ✓ Yochum and D'Amore investigated the impacts of different values for "n."
- ✓ Fatah Comlekoglu (Comlekoglu-90) investigated n-gram data structures using an inverted file system for n=2 to n=26.
- ✓ Trigrams (n-grams of length 3) were determined to be the optimal length, trading off information versus size of data structure.
- ✓ The Aquaintance System uses longer n-grams, ignoring word boundaries.
- ✓ The advantage of n-grams is that they place a finite limit on the number of searchable tokens.

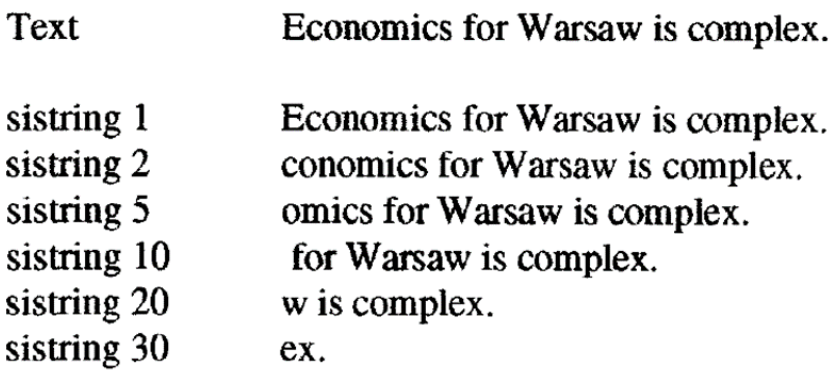
$$\mathrm{MaxSeg}_n = (\lambda)^n$$

- ✓ The maximum number of unique n-grams that can be generated, MaxSeg, can be calculated as a function of n which is the length of the n-grams, and λ which is the number of processable symbols from the alphabet (i.e., non-interword symbols).

PAT DATA STRUCTURE

- ✓ Using n-grams with interword symbols included between valid processing tokens equates to a continuous text input data structure that is being indexed in contiguous "n" character tokens.
- ✓ A different view of addressing a continuous text input data structure comes from PAT trees and PAT arrays.

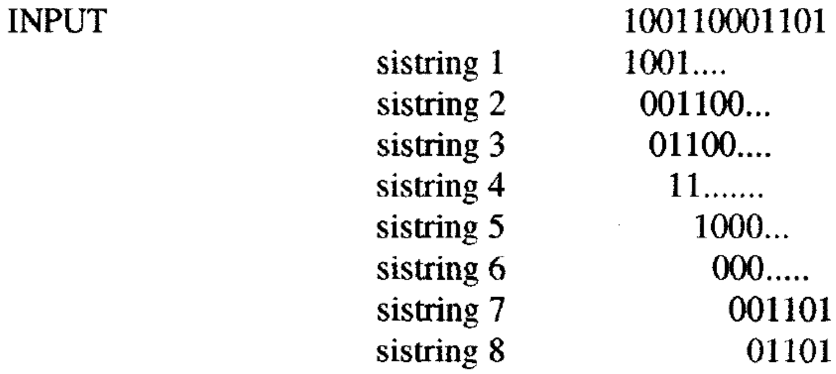
- ✓ The input stream is transformed into a searchable data structure consisting of substrings.
- ✓ The original concepts of PAT tree data structures were described as Patricia trees (Flajolet-86, Frakes-92, Gonnet-83, Knuth-73, and Morrison-68) and have gained new momentum as a possible structure for searching text and images (Gonnet-88) and applications in genetic databases (Manber-90).
- ✓ The name PAT is short for Patricia Trees (PATRICIA stands for Practical Algorithm To Retrieve Information Coded In Alphanumerics).
- ✓ In creation of PAT trees each position in the input string is the anchor point for a sub-string that starts at that point and includes all new text up to the end of the input. All substrings are unique. This view of text lends itself to many different search processing structures.
- ✓ It fits within the general architectures of hardware text search machines and parallel processors.
- ✓ A substring can start at any point in the text and can be uniquely indexed by its starting location and length.
- ✓ If all strings are to the end of the input, only the starting location is needed since the length is the difference from the location and the total length of the item.
- ✓ It is possible to have a substring go beyond the length of the input stream by adding additional null characters.
- ✓ These substrings are called sistring (semi-infinite string).
- ✓ Figure 4.9 shows some possible sistrings for an input text.



Text	Economics for Warsaw is complex.
sistring 1	Economics for Warsaw is complex.
sistring 2	conomics for Warsaw is complex.
sistring 5	omics for Warsaw is complex.
sistring 10	for Warsaw is complex.
sistring 20	w is complex.
sistring 30	ex.

Figure 4.9 Examples of sistrings

- ✓ A PAT tree is an unbalanced, binary digital tree defined by the sistrings.
- ✓ The individual bits of the sistrings decide the branching patterns with zeros branching left and ones branching right.
- ✓ PAT trees also allow each node in the tree to specify which bit is used to determine the branching via bit position or the number of bits to skip from the parent node.
- ✓ This is useful in skipping over levels that do not require branching.
- ✓ Figure 4.10 gives an example of the sistrings used in generating a PAT



INPUT		100110001101
	sistring 1	1001....
	sistring 2	001100...
	sistring 3	01100....
	sistring 4	11.......
	sistring 5	1000...
	sistring 6	000.....
	sistring 7	001101
	sistring 8	01101

Figure 4.10 Sistrings for input “100110001101”

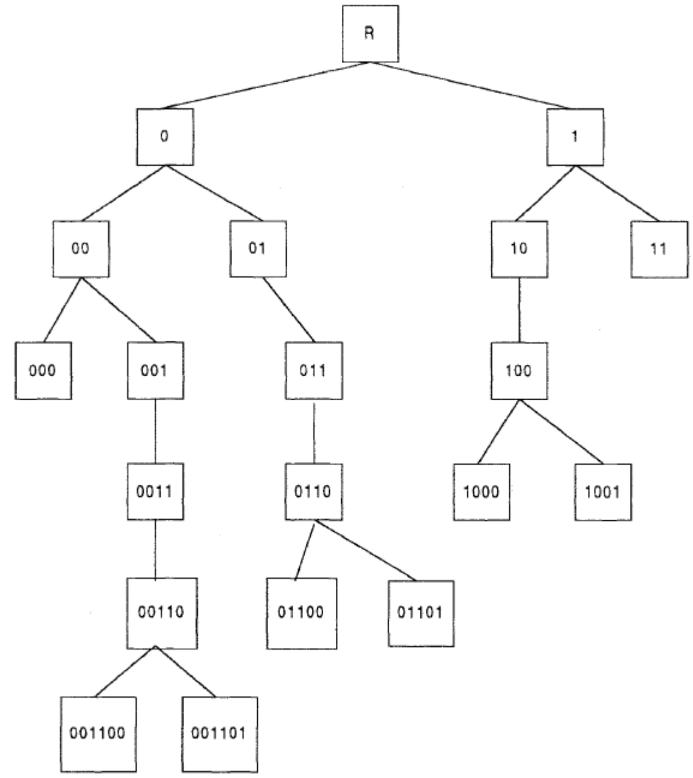


Figure 4.11 PAT Binary Tree for input "100110001101"

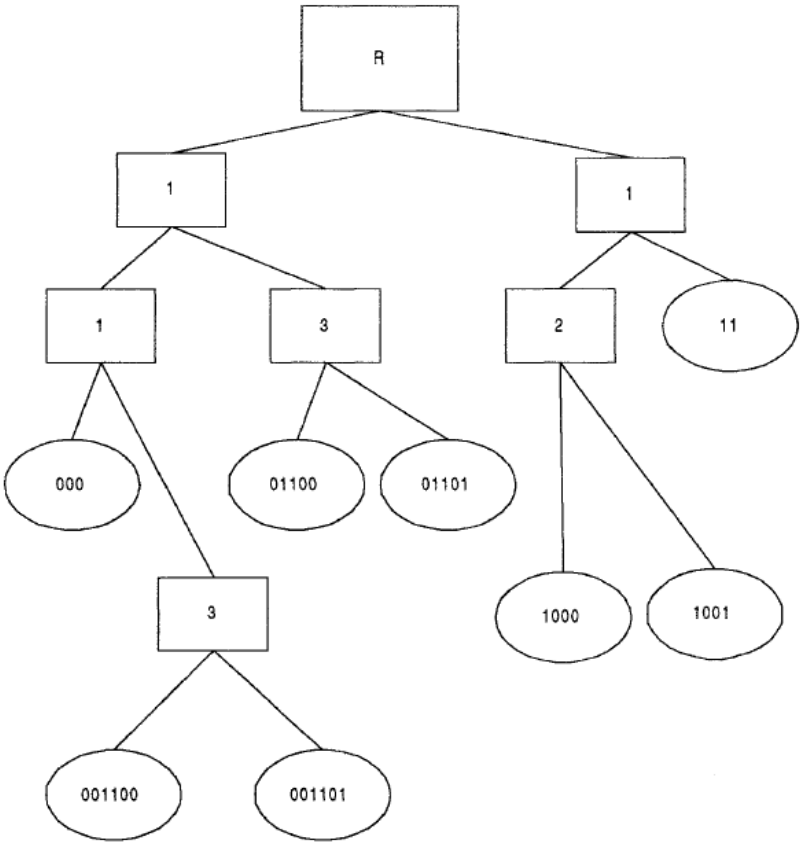
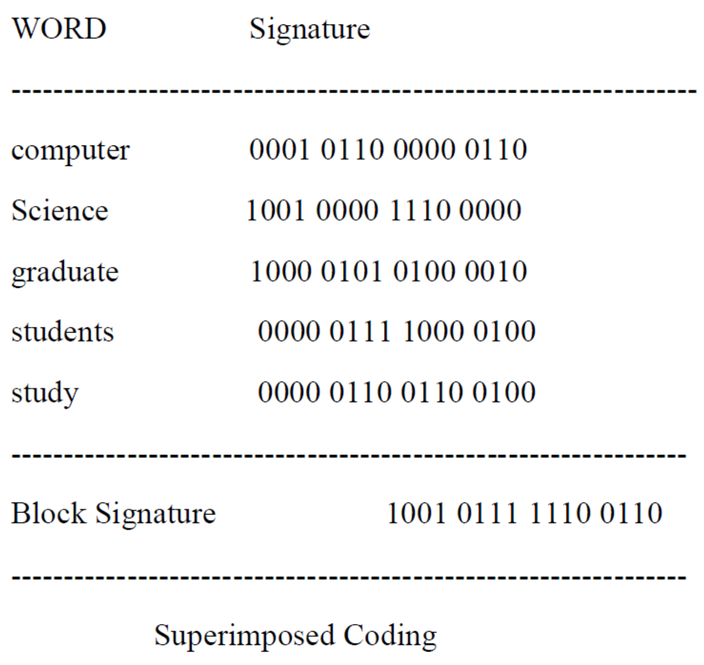


Figure 4. 12 PAT Tree skipping bits for "100110001101"

SIGNATURE FILE STRUCTURE

- ✓ The coding is based upon words in the code.
- ✓ The words are mapped into word signatures.
- ✓ A word signature is fixed length code with a fixed number of bits set to 1.
- ✓ The bit positions that are set to one are determined via a hash function of the word.
- ✓ The word signatures are Ored together to create signature of an item.
- ✓ Partitioning of words is done in block size, Which is nothing but set of words, Code length is 16 bits.
- ✓ Search is accomplished by template matching on the bit position.
- ✓ Provide a practical solution applied in parallel processing, distributed environment etc.
- ✓ To avoid signatures being too dense with 1's, a maximum number of words is specified and an item is partitioned into blocks of that size.
- ✓ The block size is set at five words, the code length is 16 bits and the number of bits that are allowed to be "1" for each word is five.
- ✓ TEXT: Computer Science graduate students study (assume block size is five words).



WORD	Signature
computer	0001 0110 0000 0110
Science	1001 0000 1110 0000
graduate	1000 0101 0100 0010
students	0000 0111 1000 0100
study	0000 0110 0110 0100
Block Signature	1001 0111 1110 0110

Superimposed Coding

Application(s)/Advantage(s)

1. Signature files provide a practical solution for storing and locating information in a number of different situations.
2. Signature files have been applied as medium size databases, databases with low frequency of terms, WORM devices, parallel processing machines, and distributed environments.

HYPERTEXT AND XML DATA STRUCTURES

- ✓ The advent of the Internet and its exponential growth and wide acceptance as a new global information network has introduced new mechanisms for representing information.
- ✓ This structure is called hypertext and differs from traditional information storage data structures in format and use.
- ✓ The hypertext is Hypertext is stored in HTML format and XML.
- ✓ Bot of these languages provide detailed descriptions for subsets of text similar to the zoning.
- ✓ Hypertext allows one item to reference another item via embedded pointer.
- ✓ HTML defines internal structure for information exchange over WWW on the internet.
- ✓ XML: defined by DTD, DOM, XSL, etc.

******

UNIT – III

CONTENTS

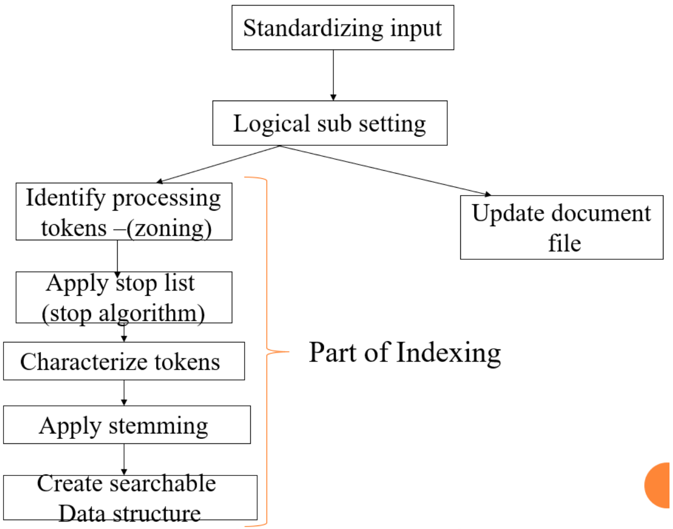
1. **Automatic Indexing:** Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing, Hypertext Linkages.
2. **Document and Term Clustering:** Introduction to Clustering, Thesaurus Generation, Item Clustering, Hierarchy of Clusters.

AUTOMATIC INDEXING

A Method of indexing in which an algorithm is applied by a computer to the title and/or text of a work to identify and extract words and phrases representing subjects, for use as headings under which entries are made in the index.

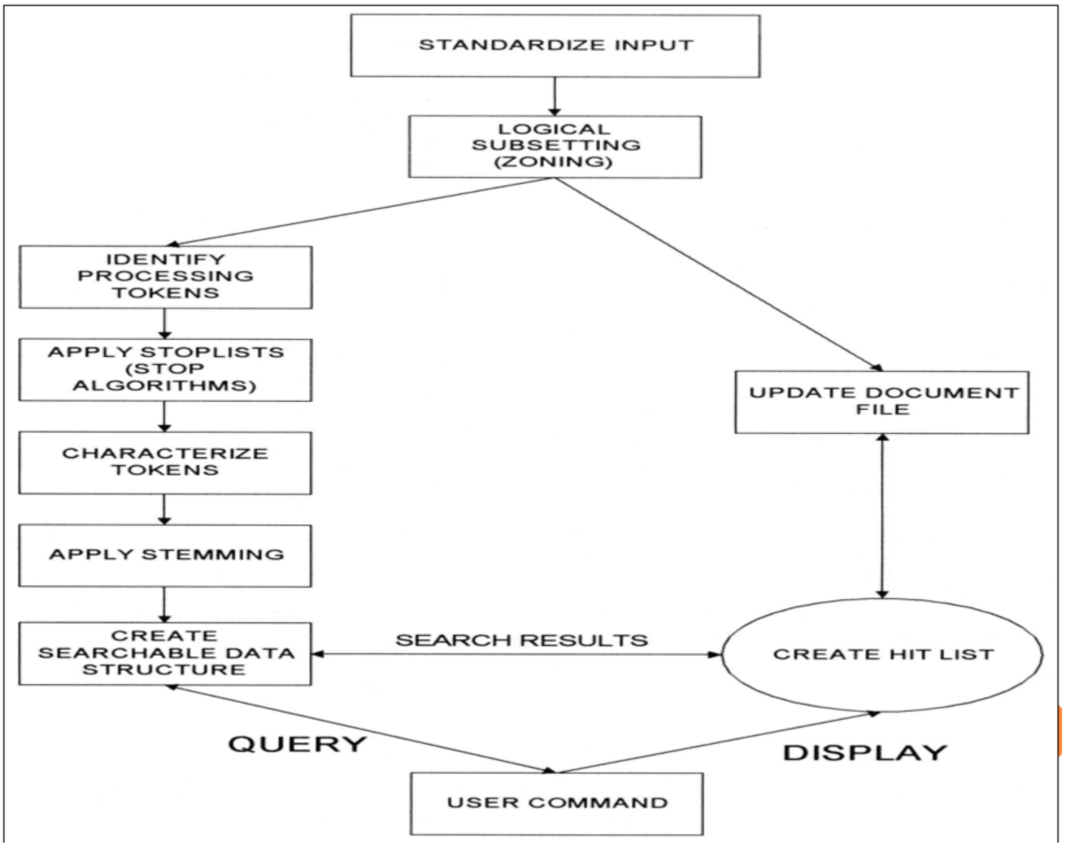
CLASSES OF AUTOMATIC INDEXING

- ✓ Automatic indexing is the process of analyzing an item to extract the information to be permanently kept in index.
- ✓ This process is associated with the generation of the searchable data structures associated with an item.
- ✓ The left hand side of the figure includes identifying processing tokens, apply stop list algorithm, characterize tokens, apply stemming, and creating searchable data structure is part of indexing process.



❖ data flow in a information processing system.

Data Flow in Information Processing System (Overall fig.)



- ✓ All systems go through the initial stage of zoning and identify the processing tokens to create index.
- ✓ Some systems automatically divide the document up into fixed length passages or localities, which become the item unit that is indexed.
- ✓ Filters such as stop list algorithm and stemming algorithms to reduce the processing tokens.
- ✓ An index is the data structure created to support search strategy.
- ✓ Search strategy – classified as statistical, natural language, and concept.
  - **Statistical strategy:** Covers the broadest range of indexing techniques and common in commercial systems.
  - **Natural language:** Approach perform the similar processing token identification as in statistical techniques - additional level of parsing of the item (present, past, future action) enhance search precision.

- **Concept:** Indexing uses the words within an item to correlate to concepts discussed in the index item.

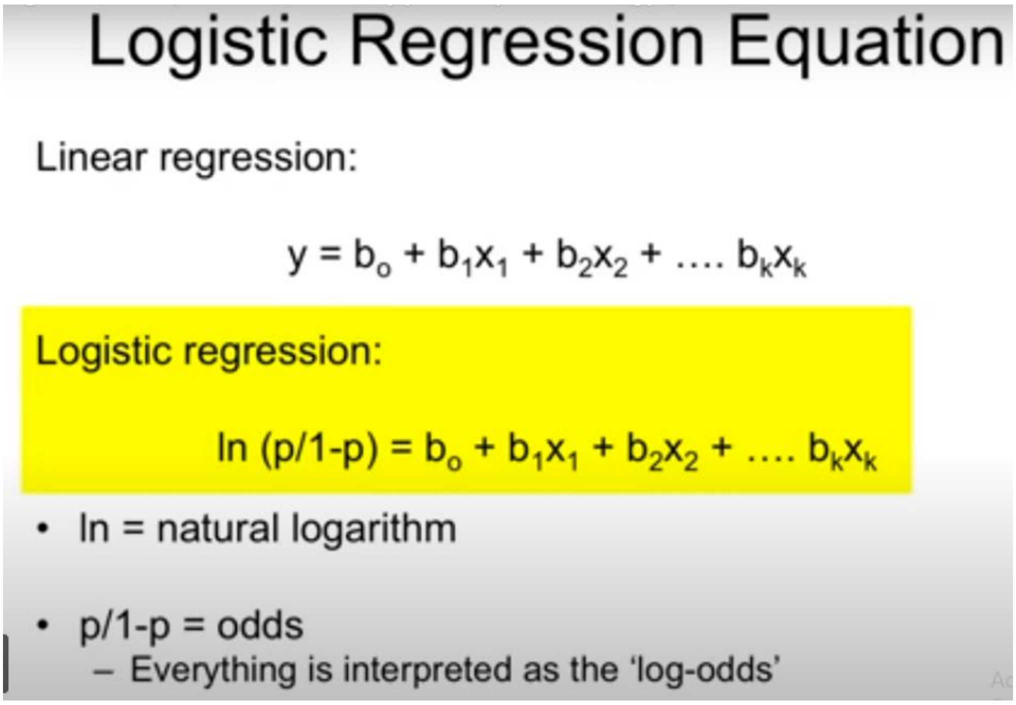
STATISTICAL INDEXING

- ✓ Uses frequency of occurrence of events to calculate the number to indicate potential relevance of an item.
- ✓ The documents are found by normal Boolean search and then statistical calculation are performed on the hit file, ranking the output (e.g. The term-frequency algorithm).
  1. Probability weighting.
  2. Vector Weighting.
     - i. Simple Term Frequency algorithm.
     - ii. Inverse Document Frequency algorithm.
     - iii. Signal Weighting.
     - iv. Discrimination Value.
     - v. Problems with the weighting schemes and vector model.
  3. Bayesian Model

1. Probabilistic weighting

- ✓ Probabilistic systems attempt to calculate a probability value that should be invariant to both calculation method and text corpora (large collection of written/spoken texts).
- ✓ The probabilistic approach is based on direct application of theory of probability to information retrieval system.
- ✓ Advantage: uses the probability theory to develop the algorithm.
- ✓ This allows easy integration of the final results when searches are performed across multiple databases and use different search algorithms.
- ✓ The use of probability theory is a natural choice – the basis of evidential reasoning (drawing conclusions from evidence).

- ✓ Advantage of probabilistic approach is that it can identify its weak assumptions and work to strengthens them.
- ✓ Ex: logistical regression**.**



Logistic Regression Equation

Linear regression:

$$y = b_o + b_1x_1 + b_2x_2 + \dots b_kx_k$$

Logistic regression:

$$\ln (p/1-p) = b_o + b_1x_1 + b_2x_2 + \dots b_kx_k$$

- ln = natural logarithm
- p/1-p = odds
  - Everything is interpreted as the 'log-odds'

- ✓ Logarithm O is the logarithm of odds of relevance for terms Tk which is present in document Dj and query Qi

$$\log(O(R \mid Q_i, D_j, t_k)) = c_0 + c_1 v_1 + \dots + c_n v_n$$

- ✓ The logarithm that the i^{th} Query is relevant to the j^{th} document is the sum of the log odds for all terms:

$$\log(O(R \mid Q_i, D_j)) = \sum_{k=1}^{q} [\log(O(R \mid Q_i, D_j, t_k)) - \log(O(R))]$$

- ✓ The inverse logistic transformation is applied to obtain the probability of relevance of a document to a query:

$$P(R \mid Q_i , D_j) = 1 \backslash (1 + e^{-\log(O(R \mid Q_i, D_j))})$$

✓ The coefficients of the equation for log odds is derived for a particular database using a random sample of query-document-term-relevance quadruples and used to predict odds of relevance for other query-document pairs.

✓ Additional attributes of relative frequency in the query (QRF), relative frequency in the document (DRF) and relative frequency of the term in all the documents (RFAD) were included, producing the log odds formula: QRF = QAF\ (total number of terms in the query), DRF = DAF\(total number of words in the document) and RFAD = (total number of term occurrences in the database)\ (total number of all words in the database).

$$Z_j = \log(O(R \mid t_j)) = c_0 + c_1\log(QAF) + c_2\log(QRF) + c_3\log(DAF) + c_4\log(DRF) + c_5\log(IDF) + c_6\log(RFAD)$$

✓ Logs are used to reduce the impact of frequency information; then smooth out skewed distributions (These distributions are sometimes called asymmetric or asymmetrical distributions).

✓ A higher max likelihood is attained for logged attributes.

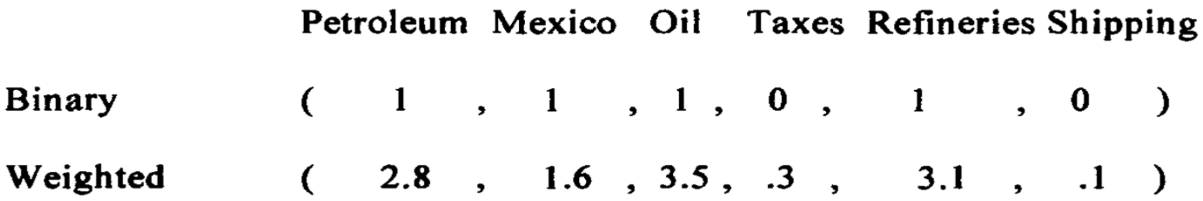
✓ The coefficients and log (O(R)) were calculated creating the final formula for ranking for query vector Q’, which contains q terms:

$$\log(O(R \mid \vec{Q})) = -5.138 + \sum_{k=1}^{q} (Z_j + 5.138)$$

2. Vector weighing

✓ Earliest system that investigated statistical approach is SMART (System for the Mechanical Analysis and Retrieval of Text) system of Cornell university. – system based on vector model.

- ✓ Vector is one dimension of set of values, where the order position is fixed and represents a domain.
- ✓ Each position in the vector represents a processing token.
- ✓ Two approaches to the domain values in the vector – binary or weighted.
- ✓ Under binary approach the domain contains a value of 1 or 0.
- ✓ Under weighted - domain is set of positive value – the value of each processing token represents the relative importance of the item.
- ✓ Binary vector requires a decision process to determine if the degree that a particular token the semantics of item is sufficient to include in the vector.
- ✓ Ex., a five-page item may have had only one sentence like "Standard taxation of the shipment of the oil to refineries is enforced."
- ✓ For the binary vector, the concepts of "Tax" and "Shipment" are below the threshold of importance (e.g., assume threshold is 1.0) and they not are included in the vector.



	Petroleum	Mexico	Oil	Taxes	Refineries	Shipping
Binary	(1 ,	1 ,	1 ,	0 ,	1 ,	0)
Weighted	(2.8 ,	1.6 ,	3.5 ,	.3 ,	3.1 ,	.1)

Fig: Binary and Vector Representation of an Item

- ✓ A Weighted vector acts same as the binary vector but provides a range of values that accommodates a variance in the value of relative importance of processing tokens in representing the item.
- ✓ The use of weights also provides a basis for determining the rank of an item.
- ✓ Each processing token can be considered another dimension in an item representation space.
- ✓ 3D vector representation assuming there were only three processing tokens, Petroleum Mexico and Oil.

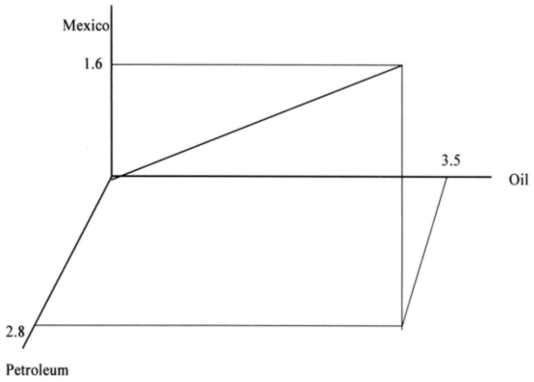


Fig: Vector Representation

i. Inverse document frequency (IDF)

- ✓ Enhancing the weighting algorithm: The weights assigned to the term should be inversely proportional to the frequency of term occurring in the data base.
- ✓ The term "computer" represents a concept used in an item, but it does not help a user find the specific information being sought since it returns the complete DB.
- ✓ This leads to the general statement enhancing weighting algorithms that the weight assigned to an item should be inversely proportional to the frequency of occurrence of an item in the database.
- ✓ This algorithm is called inverse document frequency (IDF).
- ✓ Weight calculations using inverse document frequency

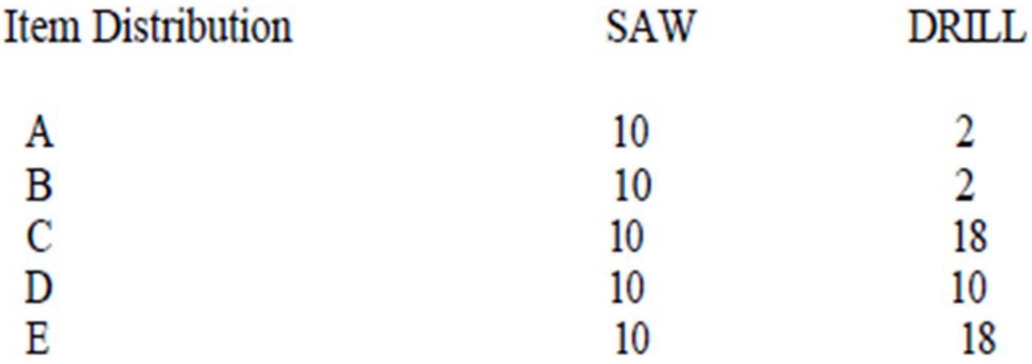
$$\text{Weight}_{\text{oil}} = 4 * (\text{Log}_2(2048) - \text{Log}_2(128) + 1) = 4 * (11 - 7 + 1) = 20$$

$$\text{Weight}_{\text{Mexico}} = 8 * (\text{Log}_2(2048) - \text{Log}_2(16) + 1) = 8 * (11 - 4 + 1) = 64$$

$$\text{Weight}_{\text{refinery}} = 10 * (\text{Log}_2(2048) - \text{Log}_2(1024) + 1) = 10 * (11 - 10 + 1) = 20$$

ii. Signal weighting

- ✓ IDF adjusts the weight of a processing token for an item based upon the number of items that contain the term in the existing database.
- ✓ It does not account for is the term frequency distribution of the processing token in the items that contain the term - can affect the ability to rank items.
- ✓ For example, assume the terms “SAW” and “DRILL” are found in 5 items with the following frequencies:



Item Distribution	SAW	DRILL
A	10	2
B	10	2
C	10	18
D	10	10
E	10	18

- ✓ Formula for calculating the weighting factor called Signal (Dennis-67) can be used:

$$\text{Signal}_k = \text{Log}_2\,(\text{TOTF}) - \text{AVE\_INFO}$$

iii. Discrimination value

- ✓ Creating a weighting algorithm based on the discrimination of value of the term.
- ✓ All items appear the same, the harder it is to identify those that are needed.
- ✓ Salton and Yang proposed a weighting algorithm that takes into consideration the ability for a search term to discriminate among items.
- ✓ They proposed use of a discrimination value for each term “i”:

$$\text{DISCRIM}_i = \text{AVESIM}_i - \text{AVESIM}$$

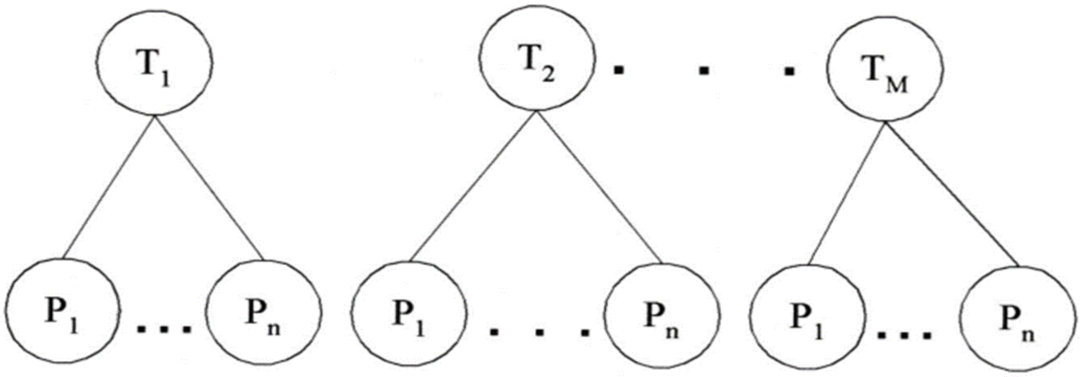
- Where AVESIM is the average similarity between every item in the database and AVESIMi is the same calculation except that term “i” is removed from all items.
- DISCRIMi value being positive, close to zero/negative.
- Once the value of DISCRMi is normalized as a positive number, it can be used in the standard weighting formula as:

$$\text{Weight}_{ik} = \text{TF}_{ik} * \text{DISCRIM}_k$$

3. Bayesian Model

- One way of overcoming the restrictions inherent in a vector model is to use a Bayesian approach to maintaining information on processing tokens.
- The Bayesian model provides a conceptually simple yet complete model for information systems.
- The Bayesian approach is based upon conditional probabilities (e.g., Probability of Event 1 given Event 2 occurred).
- This general concept can be applied to the search function as well as to creating the index to the database.
- The objective of information systems is to return relevant items.
- Thus the general case, using the Bayesian formula, is P (REL/DOCi, Queryj) which is interpreted as the probability of relevance (REL) to a search statement given a particular document and query.
- In addition to search, Bayesian formulas can be used in determining the weights associated with a particular processing token in an item.
- The objective of creating the index to an item is to represent the semantic information in the item.

- ✓ A Bayesian network can be used to determine the final set of processing tokens (called topics) and their weights.
- ✓ A simple view of the process where Ti represents the relevance of topic "i" in a particular item and Pj represents a statistic associated with the event of processing token "j" being present in the item.



NATURAL LANGUAGE

- ✓ The goal of natural language processing is to use the semantic information in addition to the statistical information to enhance the indexing of the item.
- ✓ This improves the precision of searches, reducing the number of false hits a user reviews.
- ✓ The semantic information is extracted as a result of processing the language rather than treating each word as an independent entity.
- ✓ The simplest output of this process results in generation of phrases that become indexes to an item.
- ✓ More complex analysis generates thematic representation of events rather than phrases.
- ✓ Statistical approaches use proximity as the basis behind determining the strength of word relationships in generating phrases.
- ✓ For example, with a proximity constraint of adjacency, the phrases "venetian blind" and "blind Venetian" may appear related and map to the same phrase.

✓ But syntactically and semantically those phrases are very different concepts.

✓ Word phrases generated by natural language processing algorithms enhance indexing specification and provide another level of disambiguation.

✓ Natural language processing can also combine the concepts into higher level concepts sometimes referred to as thematic representations.

1. Index Phrase Generation

✓ The goal of indexing is to represent the semantic concepts of an item in the information system to support finding relevant information.

✓ Single words have conceptual context, but frequently they are too general to help the user find the desired information.

✓ Term phrases allow additional specification and focusing of the concept to provide better precision and reduce the user's overhead of retrieving non-relevant items.

✓ Having the modifier "grass" or "magnetic" associated with the term "field" clearly disambiguates between very different concepts.

✓ One of the earliest statistical approaches to determining term phrases using of a COHESION factor between terms (Salton-83):

$$\text{COHESION}_{k,h} = \text{SIZE-FACTOR} * (\text{PAIR-FREQ}_{k,h} / \text{TOTF}_{k} * \text{TOTF}_{H})$$

2. Natural Language Processing

✓ Natural language processing not only produces more accurate term phrases, but can provide higher level semantic information identifying relationships between concepts.

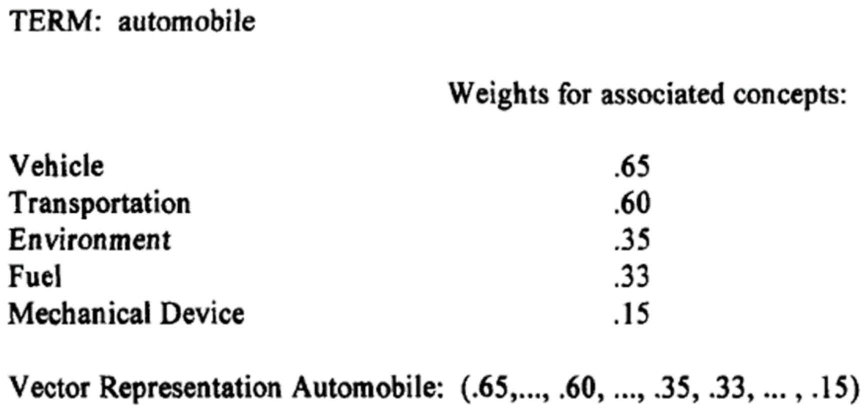
✓ System adds the functional processes Relationship Concept Detectors, Conceptual Graph Generators and Conceptual Graph Matchers that generate higher level linguistic relationships including semantic and is course level relationships.

- ✓ During the first phase of this approach, the processing tokens in the document are mapped to Subject Codes.
- ✓ These codes equate to index term assignment and have some similarities to the concept-based systems.
- ✓ The next phase is called the Text Structurer, which attempts to identify general discourse level areas within an item.
- ✓ The next level of semantic processing is the assignment of terms to components, classifying the intent of the terms in the text and identifying the topical statements.
- ✓ The next level of natural language processing identifies interrelationships between the concepts.
- ✓ The final step is to assign final weights to the established relationships.
- ✓ The weights are based upon a combination of statistical information and values assigned to the actual words used in establishing the linkages.

CONCEPT INDEXING

- ✓ Natural language processing starts with a basis of the terms within an item and extends the information kept on an item to phrases and higher level concepts such as the relationships between concepts.
- ✓ Concept indexing takes the abstraction a level further.
- ✓ Its goal is use concepts instead of terms as the basis for the index, producing a reduced dimension vector space.
- ✓ Concept indexing can start with a number of unlabeled concept classes and let the information in the items define the concepts classes created.
- ✓ A term such as “automobile” could be associated with concepts such as “vehicle,” “transportation,” “mechanical device,” “fuel,” and “environment.”

- The term "automobile" is strongly related to "vehicle," lesser to "transportation" and much lesser the other terms.
- Thus a term in an item needs to be represented by many concept codes with different weights for a particular item.
- The basis behind the generation of the concept approach is a neural network model.
- Special rules must be applied to create a new concept class.
- Example demonstrates how the process would work for the term "automobile."



TERM: automobile

	Weights for associated concepts:
Vehicle	.65
Transportation	.60
Environment	.35
Fuel	.33
Mechanical Device	.15

Vector Representation Automobile: (.65,..., .60, ..., .35, .33, ... , .15)

HYPERTEXT LINKAGES

- It's a new class of information representation is evolving on the Internet.
- Need to be generated manually Creating an additional information retrieval dimension.
- Traditionally the document was viewed as two dimensional.
- Text of the item as one dimension and references as second dimension.
- Hypertext with its linkages to additional electronic items, can be viewed as networking between the items that extend contents, i.e by embedding linkage allows the user to go immediately to the linked item.
- Issue: how to use this additional dimension to locate relevant information.
- At the internet we have three classes of mechanism to help find information.

1. **Manually generated indexes:** ex: www.yahoo.com were information sources on the home page are indexed manually into hyperlink hierarchy.
2. **Automatically generated indexes:** Sites like lycos.com and altavista.com automatically go to other internet sites and return the text , ggogle.com.
3. **Web Crawler's:** A web crawler (also known as a Web spider or Web robot) is a program or automated script which browses the World Wide Web in a methodical, automated manner.

- ✓ Are tools that that allow a user to define items of interest and they automatically go to various sites on the net and search for the desired information.' know as search tool rather than indexing.
- ✓ What is needed is an index algorithm for items that look at hypertext linkages as an extension of the concept where the link exits.
- ✓ Current concept is defined by the proximity of information.
- ✓ Attempts have been made to achieve automatically generate hyper link between items. But they suffer from the dynamic growing data bases.
- ✓ Significant portion of errors have come from parsing rather than algorithm problems.

DOCUMENT AND TERM CLUSTERING

INTRODUCTION TO CLUSTERING

- ✓ The concept of clustering has been around as long as there have been libraries.
- ✓ One of the first uses of clustering was an attempt to cluster items discussing the same subject.
- ✓ The goal of the clustering was to assist in the location of information.
- ✓ This eventually lead to indexing schemes used in organization of items in libraries and standards associated with use of electronic indexes.

- ✓ Clustering of words originated with the generation of thesauri. Thesaurus, coming from the Latin word meaning "treasure," is similar to a dictionary in that it stores words.
- ✓ Instead of definitions, it provides tile synonyms and antonyms for the words. Its primary purpose is to assist authors in selection of vocabulary.
- ✓ The goal of clustering is to provide a grouping of similar objects (e.g., terms or items) into a "class" under a more general title.
- ✓ Clustering also allows linkages between clusters to be specified.
- ✓ The process of clustering follows the following steps:
    1. Define the domain for the clustering effort. If a thesaurus is being created, this equates to determining the scope of tile thesaurus such as "medical terms." If document clustering is being performed, it is determination of the set of items to be clustered. This can be a subset of the database or the complete database. Defining the domain for the clustering identifies those objects to be used in the clustering process mid reduce tile potential for erroneous data that could induce errors in the clustering process.
    2. Once the domain is determined, determine the attributes of the objects to be clustered. If a thesaurus is being generated, determine tile specific words in the objects to be used in the clustering process. Similarly, if documents are being clustered, the clustering process may focus on specific zones within the items (e.g., Title and abstract only, main body of the item but not the references, etc.) that are to be used to determine similarity. The objective, as with the first step (a.) is to reduce erroneous associations.
    3. Determine the strength of the relationships between the attributes whose co-occurrence in objects suggest those objects should be in the same class. For thesauri this is determining which words are synonyms and the strength of their term relationships. For

documents it may be defining a similarity function based upon word co-occurrences that determine tile similarity between two items.

4. At this point, the total set of objects and the strengths of the relationships between the objects have been determined. The final step is applying some algorithm to determine tile class(s) to which each item will be assigned.

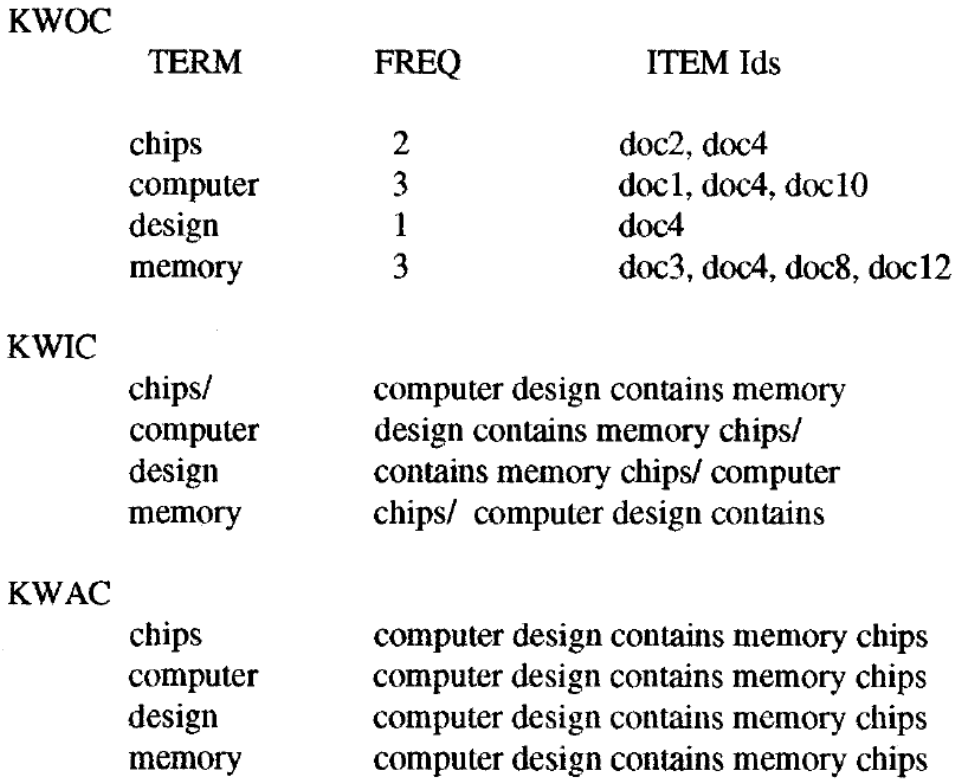
THESAURUS GENERATION

- ✓ Manual generation of clusters usually focuses on generating a thesaurus (i.e., clustering terms versus items) and has been used for hundreds of years.
- ✓ As items became available in electronic form, automated term statistical clustering techniques became available.
- ✓ Automatically generated thesauri contain classes that reflect the use of words in the corpora.
- ✓ The classes do not naturally have a name, but are just a groups of statistically similar terms.
- ✓ The optimum technique for generating the classes requires intensive computation. Other techniques starting with existing clusters can reduce the computations required but may not produce optimum classes.

Manual Clustering

- ✓ The manual clustering process follows the steps in the generation of a thesaurus.
- ✓ The first step is to determine file domain for the clustering.
- ✓ Defining the domain assists in reducing ambiguities caused by homographs and helps focus file creator.

- ✓ Usually existing thesauri, concordances from items that cover the domain and dictionaries are used as starting points for generating the set of potential words to be included in the new thesaurus.
- ✓ A concordance is an alphabetical listing of words from a set of items along with their frequency of occurrence and references of which items in which they are found.
- ✓ The art of manual thesaurus construction resides in the selection of the set of words to be included.
- ✓ Care is taken to not include words that are unrelated to the domain of the thesaurus or those that have very high frequency of occurrence and thus hold no information value (e.g., the term Computer in a thesaurus focused on data processing machines).
- ✓ If a concordance is used, other tools such as Key Word Out of Context (KWOC), Key Word In Context (KWIC) & Key Word And Context (KWAC) may help in determining useful words.



KWOC

TERM	FREQ	ITEM Ids
chips	2	doc2, doc4
computer	3	doc1, doc4, doc10
design	1	doc4
memory	3	doc3, doc4, doc8, doc12

KWIC

chips/	computer design contains memory
computer	design contains memory chips/
design	contains memory chips/ computer
memory	chips/ computer design contains

KWAC

chips	computer design contains memory chips
computer	computer design contains memory chips
design	computer design contains memory chips
memory	computer design contains memory chips

Figure 6.1 Example of KWOC, KWIC and KWAC

Automatic Term Clustering

- ✓ There are many techniques for the automatic generation of term clusters to create statistical thesauri.
- ✓ They all use as their basis the concept that the more frequently two terms co-occur in the same items, the more likely they are about the same concept.
- ✓ They differ by the completeness with which terms are correlated.
- ✓ The more complete the correlation, the higher the time and computational overhead to create the clusters.
- ✓ The most complete process computes the strength of the relationships between all combinations of the "n" unique words with an overhead of $O(n^2)$.
- ✓ Other techniques start with an arbitrary set of clusters and iterate on the assignment of terms to these clusters.
- ✓ The simplest case employs one pass of the data in creation of the clusters.
- ✓ When the number of clusters created is very large, the initial clusters may be used as a starting point to generate more abstract clusters creating a hierarchy.

Clustering Using Existing Clusters

- ✓ An alternative methodology for creating clusters is to start with a set of existing clusters.
- ✓ This methodology reduces the number of similarity calculations required to determine the clusters.
- ✓ The initial assignment of terms to the clusters is revised by revalidating every term assignment to a cluster.
- ✓ The process stops when minimal movement between clusters is detected.
- ✓ To minimize calculations, centroids are calculated for each cluster.
- ✓ A centroid is viewed in Physics as the center of mass of a set of objects.

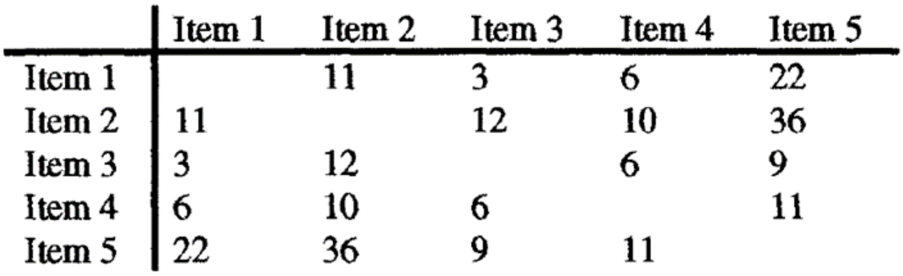
- In the context of vectors, it will equate to the average of all of the vectors in a cluster.

ITEM CLUSTERING

- Clustering of items is very similar to term clustering for the generation of thesauri.
- Manual item clustering is inherent in any library or filing system.
- In this case someone reads the item and determines the category or categories to which it belongs.
- When physical clustering occurs, each item is usually assigned to one category.
- With the advent of indexing, an item is physically stored in a primary category, but it can be found in other categories as defined by the index terms assigned to the item.
- With the advent of electronic holdings of items, it is possible to perform automatic clustering of the items.
- Using Figure 6.2 as the set of items and their terms and similarity equation:

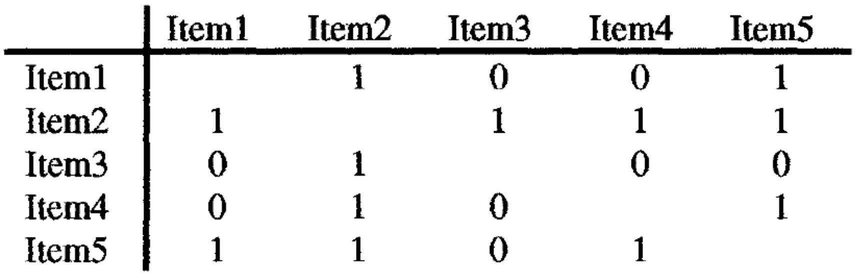
$$\mathrm{SIM}(\mathrm{Item}_i, \mathrm{Item}_j) = \Sigma\ (\mathrm{Term}_{i,k})\ (\mathrm{Term}_{j,k})$$

as k goes from 1 to 8 for the eight terms, an Item-Item matrix is created (Figure 6.9). Using a threshold of 10 produces the Item Relationship matrix shown in Figure 6.10.



	Item 1	Item 2	Item 3	Item 4	Item 5
Item 1		11	3	6	22
Item 2	11		12	10	36
Item 3	3	12		6	9
Item 4	6	10	6		11
Item 5	22	36	9	11	

Figure 6.9 Item/Item Matrix



	Item1	Item2	Item3	Item4	Item5
Item1		1	0	0	1
Item2	1		1	1	1
Item3	0	1		0	0
Item4	0	1	0		1
Item5	1	1	0	1	

Figure 6.10 Item Relationship Matrix

Using the Clique algorithm for assigning items to classes produces the following classes based upon Figure 6.10:

- ✓ Class 1 = Item 1, Item 2, Item 5
- ✓ Class 2 = Item 2, Item 3
- ✓ Class 3 = Item 2, Item 4

Application of the single link technique produces:

Class 1 = Item 1, Item 2, Item 5, Item 3, Item 4

All the items are in this one cluster, with Item 3 and Item 4 added because of their similarity to Item 2. The Star technique (i.e., always selecting the lowest non-assigned item) produces:

- ✓ Class 1 - Item 1, Item 2, Item 5
- ✓ Class 2 - Item 2, Item 3, Item 4, Item 5

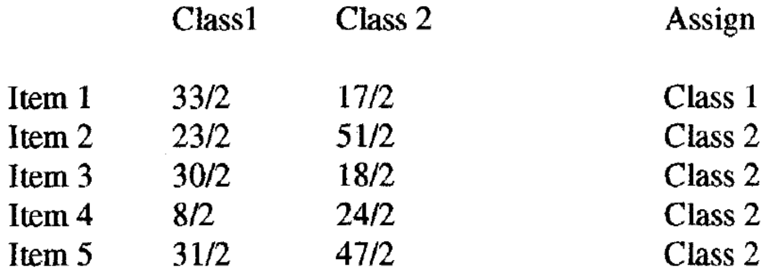
Using the String technique and stopping when all items are assigned to classes produces the following:

- ✓ Class 1 - Item 1, Item 2, Item 3
- ✓ Class 2 - Item 4, Item 5

Clustering by starting with existing clusters can be performed in a manner similar to the term model. Lets start with item 1 and item 3 in Class 1, and item 2 and item 4 in Class 2. The centroids are:

- ✓ Class 1 = 3/2, 4/2, 0/2, 0/2, 3/2, 2/2, 4/2, 3/2
- ✓ Class 2 = 3/2, 2/2, 4/2, 6/2, 1/2, 2/2, 2/2, 1/2

The results of recalculating the similarities of each item to each centroid and reassigning terms is shown in Figure 6.11.



	Class1	Class 2	Assign
Item 1	33/2	17/2	Class 1
Item 2	23/2	51/2	Class 2
Item 3	30/2	18/2	Class 2
Item 4	8/2	24/2	Class 2
Item 5	31/2	47/2	Class 2

Figure 6.11 Item Clustering with Initial Clusters

HIERARCHY OF CLUSTERS (OR) HIERARCHICAL CLUSTERING

- ✓ Hierarchical clustering in Information Retrieval focuses on the area of hierarchical agglomerative clustering methods (HACM) (Willet-88).
- ✓ The term agglomerative means the clustering process starts with unclustered items and performs pairwise similarity measures to determine tile clusters.
- ✓ Divisive is the term applied to starting with a cluster and breaking it down into smaller clusters.
- ✓ The objectives of creating a hierarchy of clusters are to:
    1. Reduce the overhead of search.
    2. Provide for a visual representation of the information space.
    3. Expand the retrieval of relevant items.
- ✓ Search overhead is reduced by performing top-down searches of the centroids of the clusters in the hierarchy and trimming those branches that are not relevant.
- ✓ It is difficult to create a visual display of the total item space.

✓ Use of dendograms along with visual cues on the size of clusters (e.g., size of the ellipse) and strengths of the linkages between clusters (e.g., dashed lines indicate reduced similarities) allows a user to determine alternate paths of browsing the database (see Figure 6.12).

✓ The dendogram allows the user to determine which clusters to be reviewed are likely to have items of interest.

✓ Even without the visual display of the hierarchy, a user can use the logical hierarchy to browse items of interest.

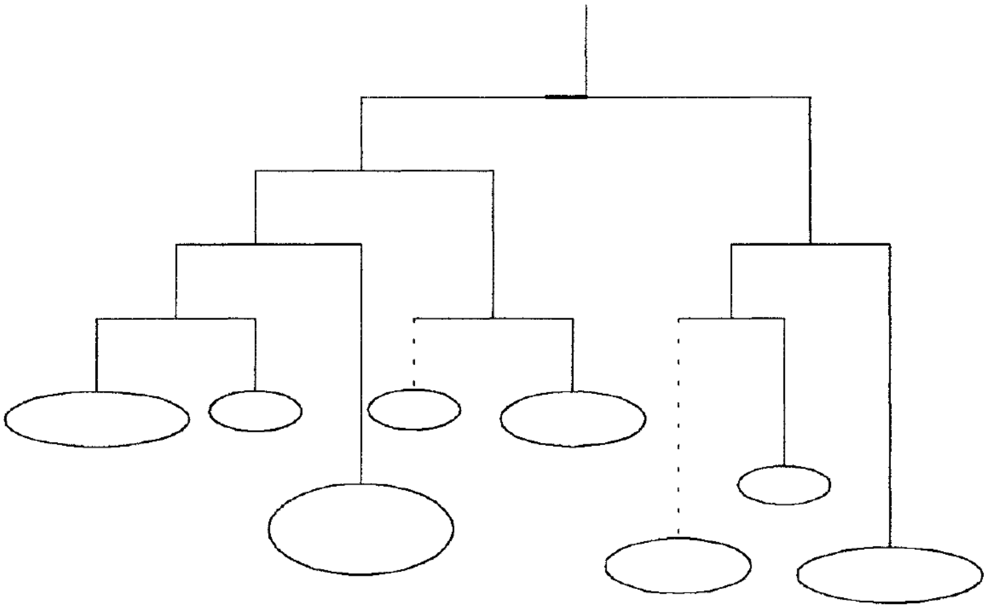


Figure 6.12 Dendogram

✓ Most of the existing HACM approaches can be defined in terms of the Lance-Williams dissimilarity update formula (Lance-66).

✓ It defines a general formula for calculating the dissimilarity D between any existing cluster Ck and a new cluster Cij created by combining clusters Ci and Cj.

$$D(C_{i,j},C_k) = \alpha_i D(C_i,C_k) + \alpha_j D(C_j,C_k) + \beta D(C_i,C_j) + \gamma | D(C_i,C_k) - D(C_j,C_k) |$$

✓ Ward's Method (Ward-63) chooses the minimum square Euclidean distance between points (e.g., centroids in this case) normalized by the number of objects in each cluster. He uses the formula for the variance I, choosing the minimum variance:

$$I_{i,j} = ((m_i m_j)/(m_i + m_{\ j}))d_{i,j}^{\ 2}$$

$$d_{i,j}^{\ 2} = \Sigma_{k=1} (x_{i,k} - x_{j,k})^2$$

******

UNIT – IV

CONTENTS

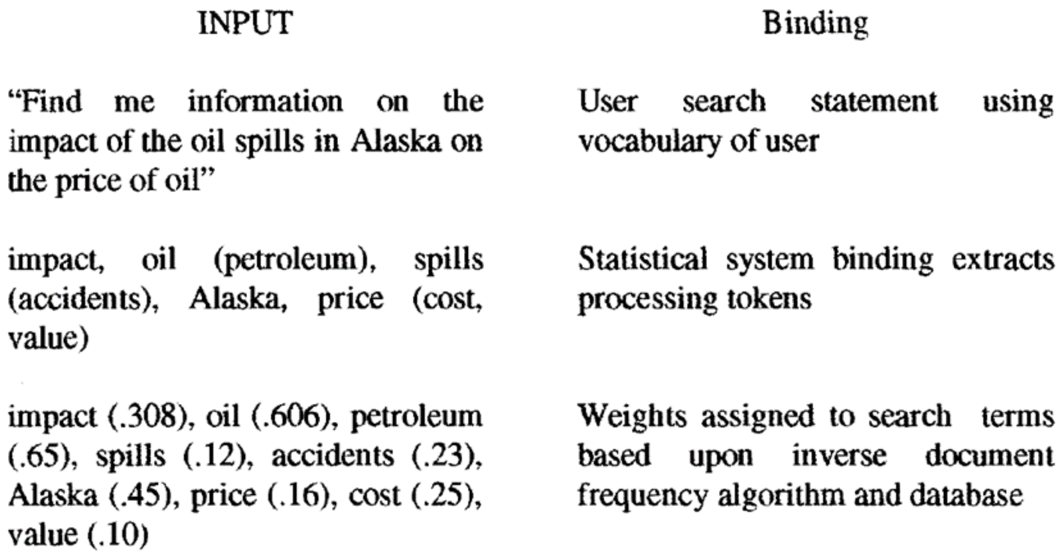
1. **User Search Techniques:** Search Statements and Binding, Similarity Measures and Ranking, Relevance Feedback, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the INTERNET and Hypertext.
2. **Information Visualization:** Introduction to Information Visualization, Cognition and Perception, Information Visualization Technologies.

USER SEARCH TECHNIQUES

SEARCH STATEMENTS AND BINDING

- ✓ Search statements are the statements of an information need generated by users to specify the concepts they are trying to locate in items.
- ✓ The search statement uses traditional Boolean logic and/or Natural Language.
- ✓ In generation of the search statement, the user may have the ability to weight (assign an importance) to different concepts in the statement.
- ✓ At this point the binding is to the vocabulary and past experiences of the user.
- ✓ Binding in this sense is when a more abstract form is redefined into a more specific form.
- ✓ The search statement is the user's attempt to specify the conditions needed to subset logically the total item space to that cluster of items that contains the information needed by the user.
- ✓ The next level of binding comes when the search statement is parsed for use by a specific search system.
- ✓ The search system translates the query to its own meta language.
- ✓ This process is similar to the indexing of item processes.

- ✓ For example, statistical systems determine the processing tokens of interest and the weights assigned to each processing token based upon frequency of occurrence from the search statement.
- ✓ Natural language systems determine the syntactical and discourse semantics using algorithms similar to those used in indexing.
- ✓ Concept systems map the search statement to the set of concepts used to index items.
- ✓ The final level of binding comes as the search is applied to a specific database.
- ✓ This binding is based upon the statistics of the processing tokens in the database and the semantics used in the database.
- ✓ This is especially true in statistical and concept indexing systems. Some of the statistics used in weighting are based upon the current contents of the database.
- ✓ Some examples are Document Frequency and Total Frequency for a specific term.
- ✓ Frequently in a concept indexing system, the concepts that are used as the basis for indexing are determined by applying a statistical algorithm against a representative sample of the database versus being generic across all databases.
- ✓ Natural Language indexing techniques tend to use the most corpora-independent algorithms.
- ✓ Figure 7.1 illustrates the three potential different levels of binding.



INPUT	Binding
"Find me information on the impact of the oil spills in Alaska on the price of oil"	User search statement using vocabulary of user
impact, oil (petroleum), spills (accidents), Alaska, price (cost, value)	Statistical system binding extracts processing tokens
impact (.308), oil (.606), petroleum (.65), spills (.12), accidents (.23), Alaska (.45), price (.16), cost (.25), value (.10)	Weights assigned to search terms based upon inverse document frequency algorithm and database

Figure 7.1 Examples of Query Binding

- ✓ Parenthesis is used in the second binding step to indicate expansion by a thesaurus.
- ✓ The length of search statements directly affect the ability of Information Retrieval Systems to find relevant items.
- ✓ The longer the search query, the easier it is for the system to find items.
- ✓ Profiles used as search statements for Selective Dissemination of Information systems are usually very long, typically 75 to 100 terms.
- ✓ In large systems used by research specialists and analysts, the typical adhoc search statement is approximately 7 terms.
- ✓ In a paper to be published in SIGIR-97, Fox et al. at Virginia Tech have noted that the typical search statement on the Internet is one or two words.
- ✓ These extremely short search statements for Search Statements.

SIMILARITY MEASURES AND RANKING

- ✓ Searching in general is concerned with calculating the similarity between a user's search statement and the items in the database.

- ✓ Restricting the similarity measure to passages gains significant precision with minimal impact on recall.
- ✓ Once items are identified as possibly relevant to the user's query, it is best to present the most likely relevant items first-Ranking is a scalar number that represents how similar an item is to the query.
- ✓ Once items are identified as possibly relevant to the user's query, it is best to present the most likely relevant items first. This process is called "ranking."
- ✓ Usually the output of the use of a similarity measure in the search process is a scalar number that represents how similar an item is to the query.

Similarity measure

- ✓ A variety of different similarity measures can be used to calculate the similarity between the item and the search statement.
- ✓ A characteristic of a similarity formula is that the results of the formula increase as the items become more similar.
- ✓ The value is zero if the items are totally dissimilar.

SIM(Itemi, Itemj) = Z(Termix) (Termja)

- ✓ This formula uses the summation of the product of the various terms of two items when treating the index as a vector.
- ✓ If Itemj is replaced with Queryj then the same formula generates the similarity between every Item and Queryj.
- ✓ The problem with this simple measure is in the normalization needed to account for variances in the length of items.
- ✓ Additional normalization is also used to have the final results come between zero and +1 (some formulas use the range -1 to +1).

- ✓ Croft expanded this original concept, taking into account the frequency of occurrence of terms within an item producing the following similarity formula (Croft-83):

Similarity formula by Salton in SMART system

To determine the "weight" an item has with respect to the search statement, the Cosine formula is used to calculate the distance between the vector for the item and the vector for the query:

$$\text{SIM}(\text{DOC}_i, \text{QUERY}_j) = \frac{\sum_{k=1}^{n} (DOC_{i,k} * QTERM_{j,k})}{\sqrt{\sum_{k=1}^{n} (DOC_{i,k})^2 * \sum_{k=1}^{n} (QTERM_{j,k})^2}}$$

The Jaccard formula is

- ✓ The denominator becomes depend upon the no of terms in common.
- ✓ Common elements common increase, the similarity value quickly decreases, in the range -1 and +1.

$$\text{SIM}(\text{DOC}_i, \text{QUERY}_j) = \frac{\sum_{k=1}^{n} (DOC_{i,k} * QTERM_{j,k})}{\sum_{k=1}^{n} DOC_{i,k} + \sum_{k=1}^{n} QTERM_{j,k} - \sum_{k=1}^{n} (DOC_{i,k} * QTERM_{j,k})}$$

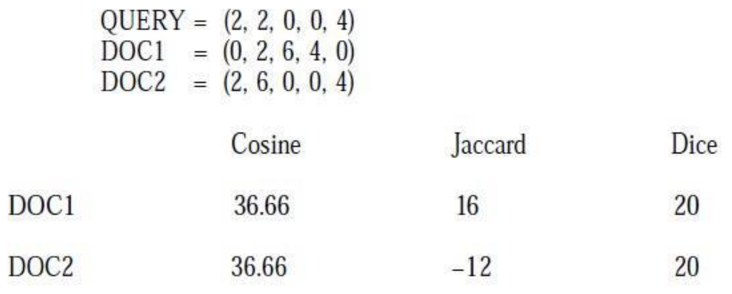
The Dice

- ✓ Measure simplifies the denominator from the Jaccard measure and introduces a factor of 2 in the numerator.
- ✓ The normalization in the Dice formula is also invariant to the number of terms in common.

$$\mathrm{SIM}(\mathrm{DOC}_i, \mathrm{QUERY}_j) = \frac{2 * \sum_{k=1}^{n} (DOC_{i,k} * QTERM_{j,k})}{\sum_{k=1}^{n} DOC_{i,k} + \sum_{k=1}^{n} QTERM_{j,k}}$$

- ✓ Use of a similarity algorithm returns the complete data base as search results.
- ✓ Many of the items have a similarity close or equal to zero.
- ✓ Thresholds (default is the similarity > zero) are usually associated with the search process.
- ✓ The threshold defines the items in the resultant Hit file from the query.
- ✓ Thresholds are either a value that the similarity measure must equal or exceed or a number that limits the number of items in the Hit file.

Normalizing denominator results vary with commonality of terms

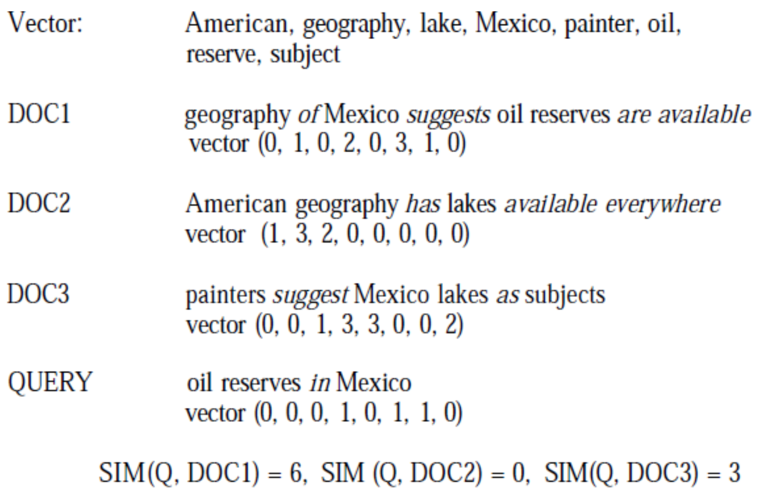


QUERY = (2, 2, 0, 0, 4)
DOC1 = (0, 2, 6, 4, 0)
DOC2 = (2, 6, 0, 0, 4)

	Cosine	Jaccard	Dice
DOC1	36.66	16	20
DOC2	36.66	–12	20

Figure 7.2 Normalizing Factors for Similarity Measures

If threshold = 4 only Doc1 is selected If threshold = 4 all selected



Vector:	American, geography, lake, Mexico, painter, oil, reserve, subject
DOC1	geography *of* Mexico *suggests* oil reserves *are available* vector (0, 1, 0, 2, 0, 3, 1, 0)
DOC2	American geography *has* lakes *available everywhere* vector (1, 3, 2, 0, 0, 0, 0, 0)
DOC3	painters *suggest* Mexico lakes *as* subjects vector (0, 0, 1, 3, 3, 0, 0, 2)
QUERY	oil reserves *in* Mexico vector (0, 0, 0, 1, 0, 1, 1, 0)

SIM(Q, DOC1) = 6, SIM (Q, DOC2) = 0, SIM(Q, DOC3) = 3

Figure 7.3 Query Threshold Process

The items are stored in clusters (TOP-DOWN) A, B, C, I – centroids and K to N, P, Q, R- ITEMS

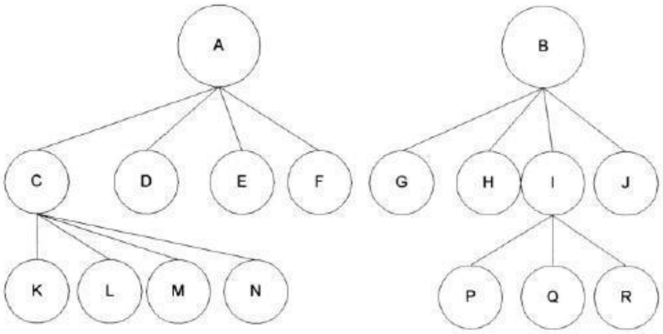


Figure 7.4 Item Cluster Hierarchy

- ✓ The query is compared to the centroids "A" and "B." If the results of the similarity measure are above the threshold, the query is then applied to the nodes' children.
- ✓ The filled circle represents the query and the filled boxes represent the centroids for the three clusters represented by the ovals.

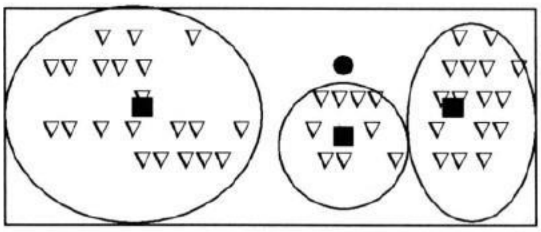


Figure 7.5 Centroid Comparisons

Ranking Algorithms

- ✓ A by-product of use of similarity measures for selecting Hit items is a value that can be used in ranking the output.
- ✓ Ranking the output implies ordering the output from most likely items that satisfy the query to least likely items.
- ✓ This reduces the user overhead by allowing the user to display the most likely relevant items first.
- ✓ The original Boolean systems returned items ordered by date of entry into the system versus by likelihood of relevance to the user's search statement.

- ✓ With the inclusion of statistical similarity techniques into commercial systems and the large number of hits that originate from searching diverse corpora, such as the Internet, ranking has become a common feature of modern systems.
- ✓ A summary of ranking algorithms from the research community is found in an article written by Belden and Croft (Belkin-87).

RELEVANCE FEEDBACK

- ✓ Thesuari and semantic networks provide utility in generally expanding a user's search statement to include potential related search terms.
- ✓ But this still does not correlate to the vocabulary used by the authors that contributes to a particular database.
- ✓ There is also a significant risk that the thesaurus does not include the latest jargon being used, acronyms or proper nouns.
- ✓ In an interactive system, users can manually modify an inefficient query or have the system automatically expand the query via a thesaurus.
- ✓ Relevant items (or portions of relevant items) are used to reweight the existing query terms and possibly expand the user's search statement with new terms.
- ✓ The relevance feedback concept was that the new query should be based on the old query modified to increase the weight of terms in relevant items and decrease the weight of terms that are in non-relevant items.
- ✓ The formula used is:

$$Q_n = Q_o + \frac{1}{r}\sum_{i=1}^{r} DR_i - \frac{1}{nr}\sum_{j=1}^{nr} DNR_j$$

where

$\mathbf{Q_n}$ = the revised vector for the new query
$\mathbf{Q_o}$ = the original query
r = number of relevant items
$\mathbf{DR_i}$ = the vectors for the relevant items
nr = number of non-relevant items
$\mathbf{DNR_j}$ = the vectors for the non-relevant items.

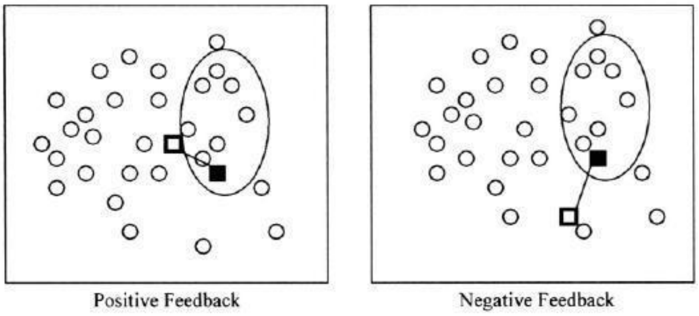
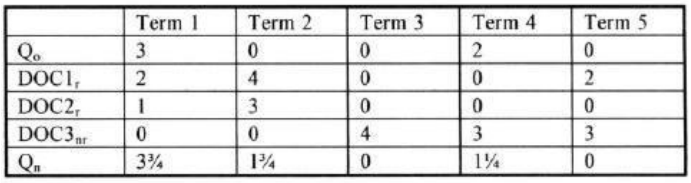
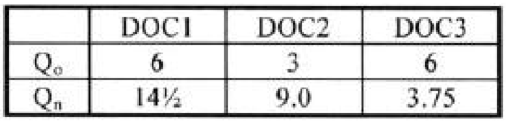


Figure 7.6 Impact of Relevance Feedback

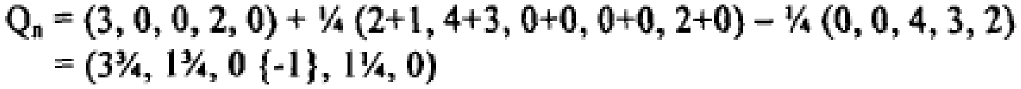


	Term 1	Term 2	Term 3	Term 4	Term 5
Q_o	3	0	0	2	0
$DOC1_r$	2	4	0	0	2
$DOC2_r$	1	3	0	0	0
$DOC3_{nr}$	0	0	4	3	3
Q_n	3¾	1¾	0	1¼	0

Figure 7.7 Query Modification via Relevance Feedback



	DOC1	DOC2	DOC3
Q_o	6	3	6
Q_n	14½	9.0	3.75



$$Q_n = (3, 0, 0, 2, 0) + \tfrac{1}{4}\,(2+1, 4+3, 0+0, 0+0, 2+0) - \tfrac{1}{4}\,(0, 0, 4, 3, 2)$$
$$= (3\tfrac{3}{4}, 1\tfrac{3}{4}, 0\ \{-1\}, 1\tfrac{1}{4}, 0)$$

SELECTIVE DISSEMINATION OF INFORMATION SEARCH

- ✓ Selective Dissemination of Information, frequently called dissemination systems, are becoming more prevalent with the growth of the Internet.
- ✓ A dissemination system is sometimes labeled a “push” system while a search system is called a “pull” system.
- ✓ In a dissemination system, the user defines a profile and as new info is added to the system it is automatically compared to the user’s profile.
- ✓ If it is considered a match, it is asynchronously sent to the user’s “mail” file.
- ✓ The differences between the two functions lie in the dynamic nature of the profiling process, the size and diversity of the search statements and number of simultaneous searches per item.
- ✓ In the search system, an existing database exists.

- ✓ As such, corpora statistics exist on term frequency within and between terms.
- ✓ These can be used for weighting factors in the indexing process and the similarity comparison (e.g., inverse document frequency algorithms).
- ✓ Profiles are relatively static search statements that cover a diversity of topics.
- ✓ Rather than specifying a particular information need, they usually generalize all of the potential information needs of a user.
- ✓ One of the first commercial search techniques for dissemination was the Logicon Message Dissemination System (LMDS). The system originated from a system created by Chase, Rosen and Wallace (CRW Inc.). It was designed for speed to support the search of thousands of profiles with items arriving every 20 seconds.
- ✓ It demonstrated one approach to the problem where the profiles were treated as the static database and the new item acted like the query.
- ✓ It uses the terms in the item to search the profile structure to identify those profiles whose logic could be satisfied by the item.
- ✓ The system uses a least frequently occurring trigraph (three character) algorithm that quickly identifies which profiles are not satisfied by the item.
- ✓ The potential profiles are analyzed in detail to confirm if the item is a hit.
- ✓ Another example of a dissemination approach is the Personal Library Software (PLS) system. It uses the approach of accumulating newly received items into the database and periodically running user's profiles against the database.
- ✓ This makes maximum use of the retrospective search software but loses near real time delivery of items.

- Another approach to dissemination uses a statistical classification technique and explicit error minimization to determine the decision criteria for selecting items for a particular profile.
- Schutze et al. used two approaches to reduce the dimensionality: selecting a set of existing features to use or creating a new much smaller set of features that the original features are mapped into.
- A x^2 measure was used to determine the most important features.
- The test was applied to a table that contained the number of relevant (N_r)and non-relevant (N^{n}_{r}) items in which a term occurs plus the number of relevant and non-relevant items in which the term does not occur respectively).
- The formula used was:

$$\chi^2 = \frac{N(N_r N_{nr-} - N_{r-} N_{nr})^2}{(N_r + N_{r-})(N_{nr} + N_{nr-})(N_r + N_{nr})(N_{r-} + N_{nr-})}$$

WEIGHTED SEARCHES OF BOOLEAN SYSTEMS

- The two major approaches to generating queries are Boolean and natural language.
- Natural language queries are easily represented within statistical models and are usable by the similarity measures discussed.
- Issues arise when Boolean queries are associated with weighted index systems.
- Some of the issues are associated with how the logic (AND, OR, NOT) operators function with weighted values and how weights are associated with the query terms.
- If the operators are interpreted in their normal interpretation, they act too restrictive or too general (i.e., AND and OR operators respectively).
- Salton, Fox and Wu showed that using the strict definition of the operators will suboptimize the retrieval expected by the user (Salton-83a).

- ✓ Closely related to the strict definition problem is the lack of ranking that is missing from a pure Boolean process.
- ✓ Some of the early work addressing this problem recognized the fuzziness associated with mixing Boolean and weighted systems (Brookstein-78, Brookstein-80).
- ✓ To integrate the Boolean and weighted systems model, Fox and Sharat proposed a fuzzy set approach (Fox-86). Fuzzy sets introduce the concept of degree of membership to a set (Zadeh-65).
- ✓ The degree of membership for AND and OR operations are defined as:

$$DEG_{A \cap B} = \min(DEG_A, DEG_B)$$

$$DEG_{A \cup B} = \max(DEG_A, DEG_B)$$

where A and B are terms in an item. DEG is the degree of membership.

- ✓ The Mixed Min and Max (MMM) model considers the similarity between query and document to be a linear combination of the minimum and maximum item weights.
- ✓ Fox proposed the following similarity formula:

$$\begin{aligned} SIM(QUERY_{OR}, DOC) &= C_{OR\,1} * \max(DOC1_1, DOC_2, \ldots, DOC_n) + \\ &\quad C_{OR2} * \min(DOC_1, DOC_2, \ldots, DOC_n) \\ SIM(QUERY_{AND}, DOC) &= C_{AND1} * \min(DOC_1, DOC_2, \ldots, DOC_n) + \\ &\quad C_{AND2} * \max(DOC1_1, DOC_2, \ldots, DOC_n) \end{aligned}$$

where C_{OR1} and C_{OR2} are weighting coefficients for the OR operation and C_{AND1} and C_{AND2} are the weighting coefficients for the AND operation. Lee and Fox found in their experiments that the best performance comes when $C_{AnD}1$ is between 0.5 to 0.8 and C_{OR1} is greater than 0.2.

- ✓ The MMM technique was expanded by Paice (Paice-84) considering all item weights versus the maximum/minimum approach. The similarity measure is calculated as:

$$\text{SIM(QUERY DOC)} = \sum_{i=1}^{n} r^{i-1} d_i \ \Big/ \ \sum_{i=1}^{n} r^{i-1}$$

where the di's are inspected in ascending order for AND queries and descending order for OR queries. The r terms are weighting coefficients. Lee and Fox showed that the best values for r are 1.0 for AND queries and 0.7 for OR queries (Lee-88).

- ✓ This technique requires more computation since the values need to be stored in ascending or descending order and thus must be sorted.
- ✓ The generalized queries are:

$$Q_{OR} = (A_{1,},a_1) \text{ OR } (A_2,a_2) \text{ OR } \ldots \text{OR } (A_n,a_n)$$

$$Q_{AND} = (A_{1,},a_1) \text{ AND } (A_2,a_2) \text{ AND } \ldots \text{ AND } (A_n,a_n)$$

SEARCHING THE INTERNET AND HYPERTEXT

- ✓ The Internet has multiple different mechanisms that are the basis for search of items.
- ✓ The primary techniques are associated with servers on the Internet that create indexes of items on the Internet and allow search of them.
- ✓ Some of the most commonly used nodes are YAHOO, AltaVista and Lycos.
- ✓ In all of these systems there are active processes that visit a large number of Internet sites and retrieve textual data which they index.

- ✓ The primary design decisions are on the level to which they retrieve data and their general philosophy on user access.
- ✓ LYCOS (http://www.lycos.com) and AltaVista automatically go out to other Internet sites and return the text at the sites for automatic indexing (http://www.altavista.digital.com).
- ✓ Lycos returns home pages from each site for automatic indexing while Altavista indexes all of the text at a site.
- ✓ The algorithm is kept relatively simple using statistical information on the occurrence of words within the retrieved text.
- ✓ Closely associated with the creation of the indexes is the technique for accessing nodes on the Internet to locate text to be indexed.
- ✓ This search process is also directly available to users via Intelligent Agents.
- ✓ Intelligent Agents provide the capability for a user to specify an information need which will be used by the Intelligent Agent as it independently moves between Internet sites locating information of interest. There are six key characteristics of intelligent agents (Heilmann-96):
    1. **Autonomy:** The search agent must be able to operate without interaction with a human agent. It must have control over its own internal states and make independent decisions. This implies a search capability to traverse information sites based upon pre-established criteria collecting potentially relevant information.
    2. **Communications Ability:** The agent must be able to communicate with the information sites as it traverses them. This implies a universally accepted language defining the external interfaces (e.g., Z39.50).
    3. **Capacity for Cooperation:** This concept suggests that intelligent agents need to cooperate to perform mutually beneficial tasks.

4. **Capacity for Reasoning:** There are three types of reasoning scenarios (Roseler-94):
    i. **Rule-based:** Where user has defined a set of conditions and actions to be taken.
    ii. **Knowledge-based:** Where the intelligent agents have stored previous conditions and actions taken which are used to deduce future actions.
    iii. **Artificial evolution based:** Where intelligent agents spawn new agents with higher logic capability to perform its objectives.
5. Adaptive Behavior: Closely tied to 1 and 4, adaptive behavior permits the intelligent agent to assess its current state and make decisions oil the actions it should take.
6. Trustworthiness: The user must trust that the intelligent agent will act on the user's behalf to locate information that the user has access to and is relevant to the user.

INFORMATION VISUALIZATION

INTRODUCTION TO INFORMATION VISUALIZATION

- ✓ The beginnings of the theory of visualization began over 2400 years ago.
- ✓ The philosopher Plato discerned that we perceive objects through the senses, using the mind.
- ✓ Our perception of the real world is a translation from physical energy from our environment into encoded neural signals.
- ✓ The mind is continually interpreting and categorizing our perception of our surroundings.
- ✓ Use of a computer is another source of input to the mind's processing functions.

- ✓ Text-only interfaces reduce the complexity of the interface but also restrict use of the more powerful information processing functions the mind has developed since birth.
- ✓ Information visualization is a relatively new discipline growing out of the debates in the 1970s on the way the brain processes and uses mental images.
- ✓ It required significant advancements in technology and information retrieval techniques to become a possibility.
- ✓ One of the earliest researchers in information visualization was Doyle, who in 1962 discussed the concept of "semantic road maps" that could provide a user a view of the whole database (Doyle-62).
- ✓ The road maps show the items that are related to a specific semantic theme.
- ✓ The user could use this view to focus his query on a specific semantic portion of the database.
- ✓ The concept was extended in the late 1960s, emphasizing a spatial organization that maps to the information in the database (Miller-68).
- ✓ Sammon implemented a non-linear mapping algorithm that could reveal document associations providing the information required to create a road map or spatial organization (Sammons-69).
- ✓ In the 1990s technical advancements along with exponential growth of available information moved the discipline into practical research and commercialization. Information visualization techniques have the potential to significantly enhance the user's ability to minimize resources expended to locate needed information.
- ✓ The way users interact with computers changed with the introduction of user interfaces based upon Windows, Icons, Menus, and Pointing devices (WIMPs).
- ✓ There are many areas that information visualization and presentation can help the user:

1. Reduce the amount of time to understand the results of a search and likely clusters of relevant information.
2. Yield information that comes from the relationships between items versus treating each item as independent.
3. Perform simple actions that produce sophisticated information search functions.

COGNITION AND PERCEPTION

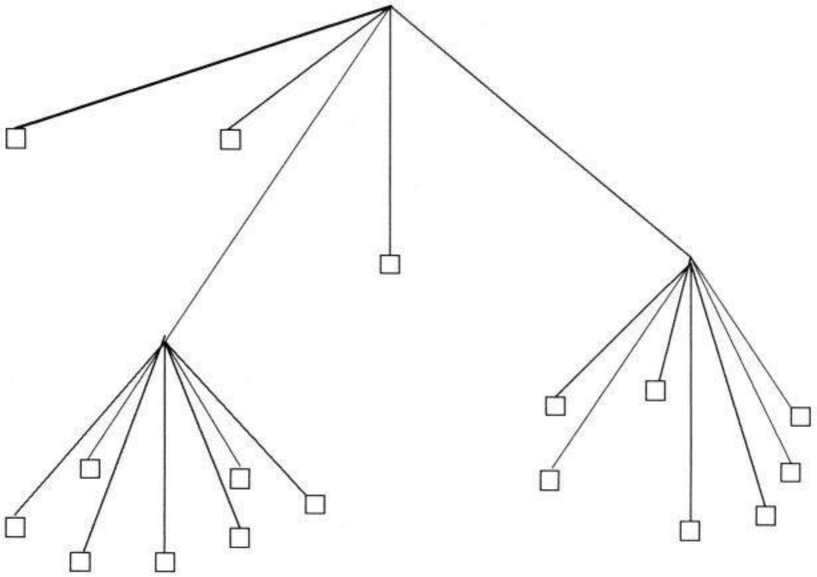
- ✓ The user-machine interface has primarily focused on a paradigm of a typewriter.
- ✓ As computers displays became ubiquitous, man-machine interfaces focused on treating the display as an extension of paper with the focus on consistency of operations.
- ✓ The advent of WIMP interfaces and the evolution of the interface focused on how to represent to the user what is taking place in the computer environment.
- ✓ Extending the HCI to improve the information flow, thus reducing wasted user overhead in locating needed information.
- ✓ Although the major focus is on enhanced visualization of information, other senses are also being looked at for future interfaces.
- ✓ The audio sense has always been part of simple alerts in computers.
- ✓ The sounds are now being replaced by speech in both input and output interfaces.
- ✓ The tactile (touch) sense is being addressed in the experiments using Virtual Reality (VR).
- ✓ A significant portion of the brain is devoted to vision and supports the maximum information transfer function from the environment to a human being.

- ✓ Visualization is the transformation of information into a visual form which enables the user to observe and understand the information.
- ✓ The Gestalt psychologists postulate that the mind follows a set of rules to combine the input stimuli to a mental representation that differs from the sum of the individual inputs (Rock- 90):
    - **Proximity:** Nearby figures are grouped together.
    - **Similarity:** Similar figures are grouped together**.**
    - **Continuity:** Figures are interpreted as smooth continuous patterns rather than discontinuous concatenations of shapes (e.g., a circle with its diameter drawn is perceived as two continuous shapes, a circle and a line, versus two half circles concatenated together).
    - **Closure:** Gaps within a figure are filled in to create a whole (e.g., using dashed lines to represent a square does not prevent understanding it as a square).
    - **Connectedness:** Uniform and linked spots, lines or areas are perceived as a single unit.

INFORMATION VISUALIZATION TECHNOLOGIES

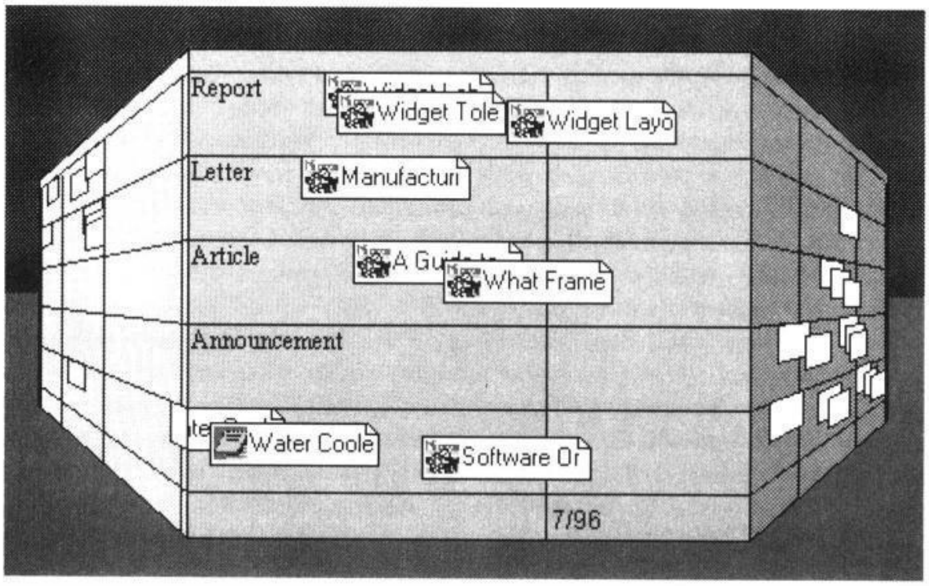
- ✓ The ones focused on Information Retrieval Systems are investigating how best to display the results of searches, structured data from DBMSs and the results of link analysis correlating data.
- ✓ The goals for displaying the result from searches fall into two major classes: document clustering and search statement analysis.
- ✓ Visualization tools in this area attempt to display the clusters, with an indication of their size and topic, as a basis for users to navigate to items of interest.
- ✓ Displaying the total set of terms, including additional terms from relevance feedback or thesaurus expansion, along with documents retrieved and indicate the importance of the term to the retrieval and ranking process.

- ✓ One way of organizing information is hierarchical.
- ✓ A tree structure is useful in representing information that ranges over time (e.g., genealogical lineage), constituents of a larger unit (e.g., organization structures, mechanical device definitions) and aggregates from the higher to lower level (e.g., hierarchical clustering of documents).
- ✓ A two-dimensional representation becomes difficult for a user to understand as the hierarchy becomes large.
- ✓ One of the earliest experiments in information visualization was the Information Visualizer developed by XEROX PARC.
- ✓ It incorporates various visualization formats such as DataMap, InfoGrid, ConeTree, and the Perspective wall.
- ✓ The Cone-Tree is a 3-Dimensional representation of data, where one node of the tree is represented at the apex and ail the information subordinate to it is arranged in a circular structure at its base.
- ✓ Any child node may also be the parent of another cone.
- ✓ Selecting a particular node rotates it to the front of the display.

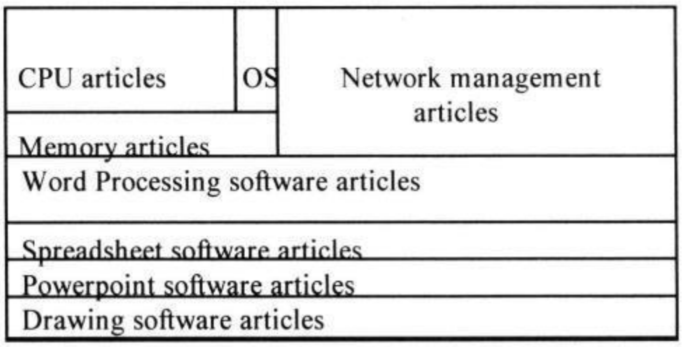


- ✓ The perspective wall divides the information into three visual areas with the area being focused on in the front and other areas out of focus to each.

- ✓ This allows the user to keep all of the information in perspective while focusing on a particular area.

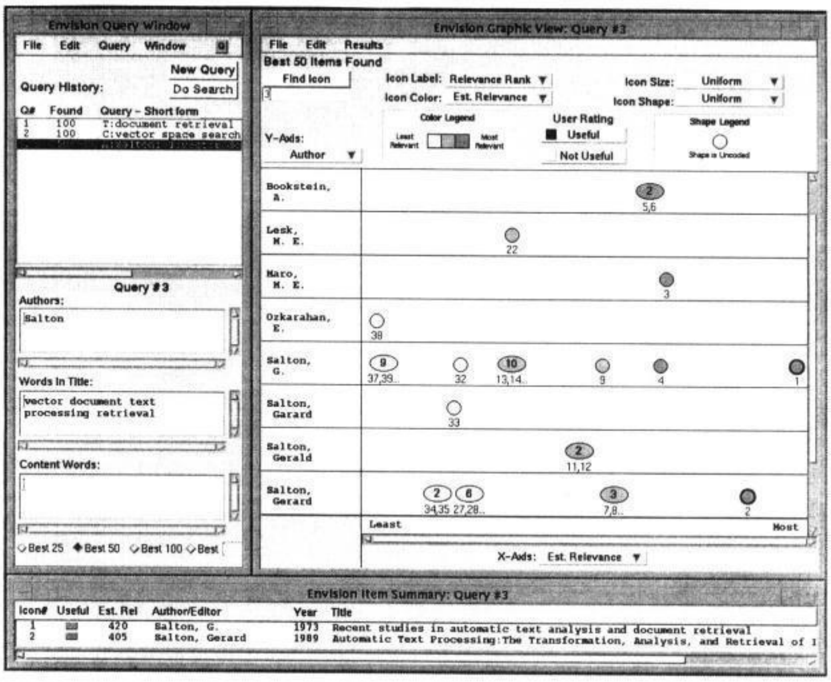


- ✓ **Tree maps (Johnson-91):** This technique makes maximum use of the display screen space by using rectangular boxes that are recursively subdivided based upon parent-child relationships between the data.
- ✓ The CPU, OS, Memory, and Network management articles are all related to a general category of computer operating systems versus computer applications which are shown in the rest of the figure.


CPU articles
OS
Network management articles
Memory articles
Word Processing software articles
Spreadsheet software articles
Powerpoint software articles
Drawing software articles

- ✓ The Envision system not only displays the relevance rank and estimated relevance of each item found by a query, but also simultaneously presents other query information.

- ✓ The design is intentionally graphical and simple using two dimensional visualization.
- ✓ This allows a larger variety of user computer platforms to have access to their system (Nowell-96).
- ✓ Envision's three interactive windows to display search results: Query window, Graphic View window, and Item Summary window.



Q#	Found	Query – Short form
1	100	T:document retrieval
2	100	C:vector space search

Authors: Salton

Words In Title: vector document text processing retrieval

Content Words:

○ Best 25 ◆ Best 50 ○ Best 100 ○ Best

Best 50 Items Found

Find Icon: 3

Icon Label: Relevance Rank

Icon Color: Est. Relevance

Icon Size: Uniform

Icon Shape: Uniform

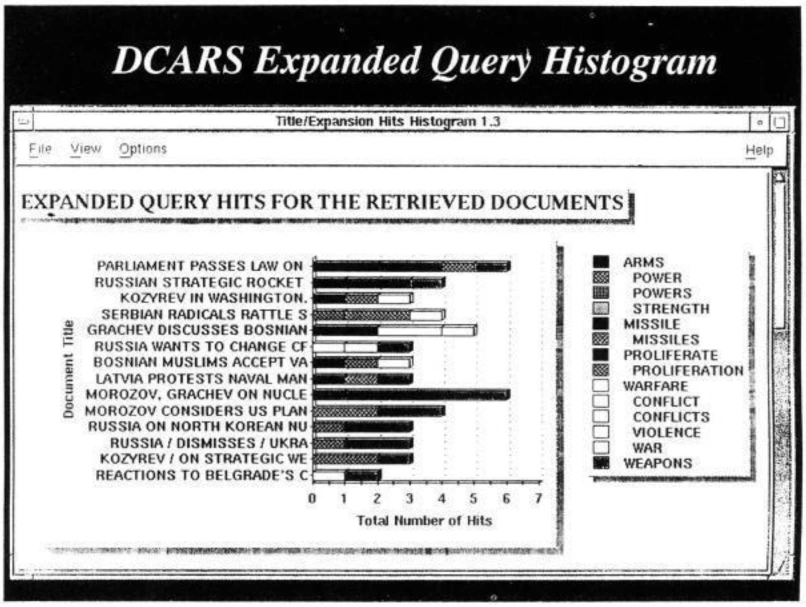
Y-Axis: Author

X-Axis: Est. Relevance

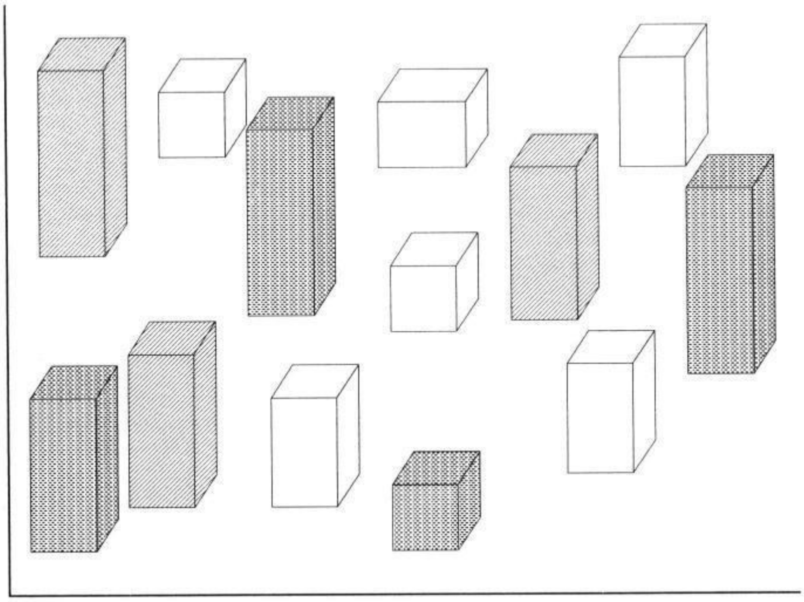
Bookstein, A.
Lesk, M. E.
Maro, M. E.
Orkarahan, E.
Salton, G.
Salton, Garard
Salton, Gerald
Salton, Gerard

Icon#	Useful	Est. Rel	Author/Editor	Year	Title
1	■	420	Salton, G.	1973	Recent studies in automatic text analysis and document retrieval
2	■	405	Salton, Gerard	1989	Automatic Text Processing:The Transformation, Analysis, and Retrieval of I

- ✓ Document Content Analysis and Retrieval System (DCARS) being developed by Calspan Advanced Technology Center.
- ✓ Their system is designed to augment the Retrieval Ware search product.
- ✓ They display the query results as a histogram with the items as rows and each term's contribution to the selection indicated by the width of a tile bar on the row.
- ✓ DCARS provides a friendly user interface that indicates why a particular item was found, but it is much harder to use the information in determining how to modify search statements to improve them.


DCARS Expanded Query Histogram
Title/Expansion Hits Histogram 1.3
File
View
Options
Help
EXPANDED QUERY HITS FOR THE RETRIEVED DOCUMENTS
Document Title
PARLIAMENT PASSES LAW ON
RUSSIAN STRATEGIC ROCKET
KOZYREV IN WASHINGTON.
SERBIAN RADICALS RATTLE S
GRACHEV DISCUSSES BOSNIAN
RUSSIA WANTS TO CHANGE CF
BOSNIAN MUSLIMS ACCEPT VA
LATVIA PROTESTS NAVAL MAN
MOROZOV, GRACHEV ON NUCLE
MOROZOV CONSIDERS US PLAN
RUSSIA ON NORTH KOREAN NU
RUSSIA / DISMISSES / UKRA
KOZYREV / ON STRATEGIC WE
REACTIONS TO BELGRADE'S C
0
1
2
3
4
5
6
7
Total Number of Hits
ARMS
POWER
POWERS
STRENGTH
MISSILE
MISSILES
PROLIFERATE
PROLIFERATION
WARFARE
CONFLICT
CONFLICTS
VIOLENCE
WAR
WEAPONS

- ✓ Another representation that is widely used for both hierarchical and network related information is the "cityscape" which uses the metaphor of movement within a city.
- ✓ In lieu of using hills, as in the terrain approach, skyscrapers represent the theme (concept) areas



******

UNIT – V

TEXT SEARCH ALGORITHMS

CONTENTS

- ✓ Introduction to Text Search Techniques.
- ✓ Software Text Search Algorithms.
- ✓ Hardware Text Search Systems.

INTRODUCTION TO TEXT SEARCH TECHNIQUES

- ✓ The basic concept of a text scanning system is the ability for one or more users to enter queries with the text of the items to be searched sequentially accessed and compared to the query terms.
- ✓ When all of the text has been accessed, the query is complete.
- ✓ One advantage of this type architecture is that as soon as an item is identified as satisfying a query, the results can be presented to the user for retrieval.
- ✓ Figure 9.1 provides a general diagram of a text streaming search system.
- ✓ The database contains the full text of the items.
- ✓ The term detector is the special hardware/software that contains all of the search terms and in some systems the logic between the terms.

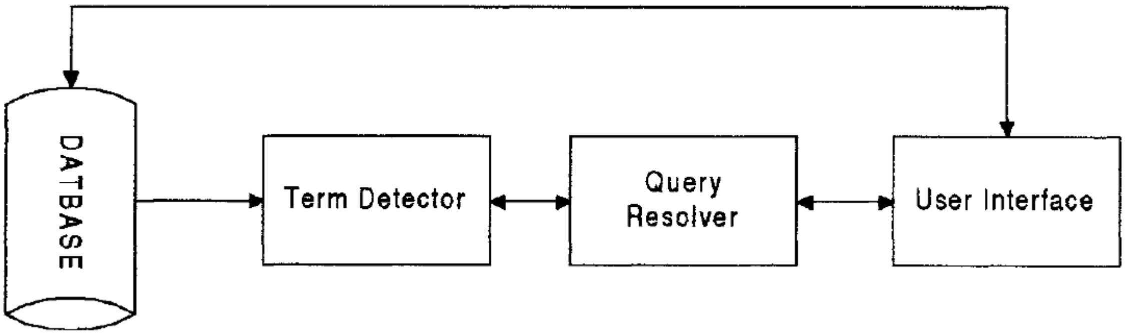


Figure 9.1 Text Streaming Architecture

- ✓ It inputs the text and detects the existence of the search terms.

- ✓ It outputs to the query resolver the detected terms, allowing for final logical processing of a query against an item.
- ✓ The query resolver performs two major functions: accepting search statements from the users and extracting the logic and search terms to pass to the detector.
- ✓ The text streaming process is focused on finding at least one or all occurrences of a pattern of text (query term) in a text stream.
- ✓ It is assumed that the same alphabet is used in both search terms and text being streamed.
- ✓ In foreign language streamers, different encodings may have to be available for items from the same language (e.g., in Cyrillic there are over six encodings that can be used).
- ✓ The worst case search for a pattern of m characters in a string of n characters is at least n - m + 1 or a magnitude of O(n) (Rivest-77).
- ✓ Some of the original brute force methods could require O(n*m) symbol comparisons (Sedgewick-88). More recent improvements have reduced the time to O(n + m).
- ✓ There are two approaches to the data stream. In the first approach the complete database is being sent to the detector(s) which function as a search of the database.
- ✓ In the second approach, random retrieved items are being passed to the detectors.
- ✓ In this second case, an index search is performed that constrains the items from the database requiring additional processing, while the text streamer performs the additional search logic that is not satisfied by the index search (Bird-78, Hollar-79).
- ✓ Examples where index searches may not be able to satisfy the complete search statement are:
  - Search for stop words.

- Search for exact matches when stemming is performed.
- Search for terms that contain both leading and trailing "don't cares".
- Search for symbols that are on the interword symbol list (e.g., ," ;).
- Search for "fuzzy" search terms

SOFTWARE TEXT SEARCH ALGORITHMS

- ✓ In software streaming techniques, the item to be searched is read into memory, and then the algorithm is applied.
- ✓ Although nothing in the described architecture prohibits software streaming from being applied to many simultaneous searches against the same item, it is more frequently used to resolve a particular search against a particular item.
- ✓ There are four major algorithms associated with software text search: the brute force approach, Knuth-Morris-Pratt, Boyer-Moore, Shift-OR algorithm, and Rabin-Karp.
- ✓ Of all of the algorithms, Boyer-Moore has been the fastest, requiring at most O(n + m) comparisons (Smit-82) where n is the number of characters being searched and rn is the size of the search string. Knuth-Pratt-Morris and Boyer-Moore both require O(n) preprocessing of search strings in addition to the search comparisons (Knuth-77, Boyer-77, Rytter-80).
- ✓ The brute force approach is the simplest string matching algorithm.
- ✓ The idea is to match the search string against the input text.
- ✓ Whenever a mis-match is detected in the comparison process, the input text is shifted one position, and the comparison process is initialized and restarted.
- ✓ The expected number of comparisons when searching an input text string of n characters for a pattern of m characters is (Baeza-Yates-89):

$$N_c = \frac{c}{c-1}(1 - {}^{1}\!/_{c^m})(n - m + 1) + O(1)$$

where N_c is the expected number of comparisons and c is the size of the alphabet for the text.

Applying the formula to an example where the alphabet is c=25 characters, the search pattern is m=7 characters, and the search stream is an item with n=30,000 characters has the number of comparisons is:

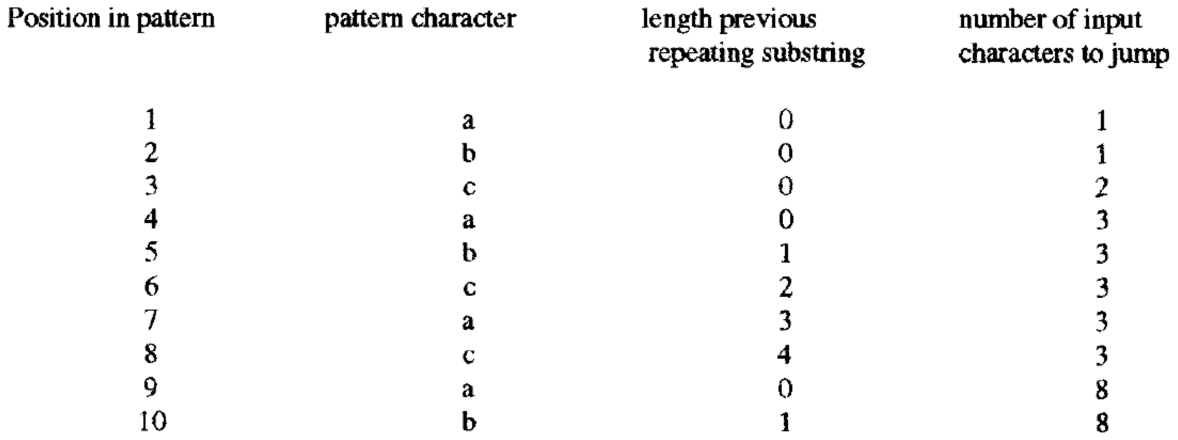
$$N_c = 25/(25 - 1)\ (1 - 1/25^7)(30000 - 7 + 1) = (1.04)(1\text{-}0)(29996) \approx n$$

- ✓ For search of any large streams the number of comparisons can be estimated by the number of characters being searched. For smaller items the length of the text pattern (m) can have an effect on the number of comparisons.
- ✓ The Knuth-Pratt-Morris algorithm made a major improvement in previous algorithms in that even in the worst case it does not depend upon the length of the search term and does not require comparisons for every character in the input stream.
- ✓ The basic concept behind the algorithm is that whenever a mismatch is detected, the previous matched characters define the number of characters that can be skipped in the input stream prior to starting the comparison process again. For example consider:

Position		1	2	3	4	5	6	7	8
Input Stream	=	a	b	d	a	d	e	f	g
Search Pattern	=	a	b	d	f				

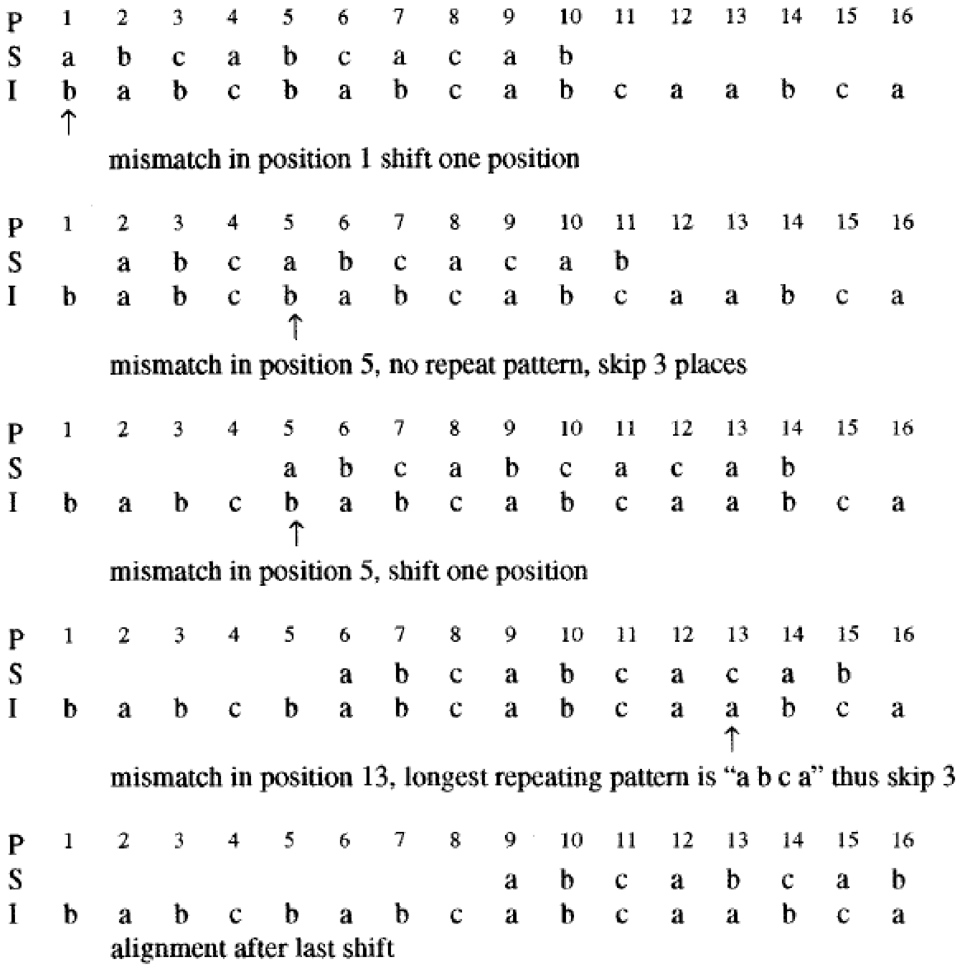
- ✓ The Shift Table that specifies the number of places to jump given a mismatch is shown in Figure 9.3 for a search pattern = abcabcacab.
- ✓ In tile table it should be noted that the alignment is primarily based on aligning over the repeats of the letters "a" and "ab."

- ✓ Figure 9.4 provides an example application of the algorithm (Salton-89) where S is the search pattern and I is the input text stream.



Position in pattern	pattern character	length previous repeating substring	number of input characters to jump
1	a	0	1
2	b	0	1
3	c	0	2
4	a	0	3
5	b	1	3
6	c	2	3
7	a	3	3
8	c	4	3
9	a	0	8
10	b	1	8

Figure 9.3 Shift Characters Table



P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
S	a	b	c	a	b	c	a	c	a	b						
I	b	a	b	c	b	a	b	c	a	b	c	a	a	b	c	a
	↑															

mismatch in position 1 shift one position

P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
S		a	b	c	a	b	c	a	c	a	b					
I	b	a	b	c	b	a	b	c	a	b	c	a	a	b	c	a
					↑											

mismatch in position 5, no repeat pattern, skip 3 places

P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
S					a	b	c	a	b	c	a	c	a	b		
I	b	a	b	c	b	a	b	c	a	b	c	a	a	b	c	a
					↑											

mismatch in position 5, shift one position

P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
S						a	b	c	a	b	c	a	c	a	b	
I	b	a	b	c	b	a	b	c	a	b	c	a	a	b	c	a
													↑			

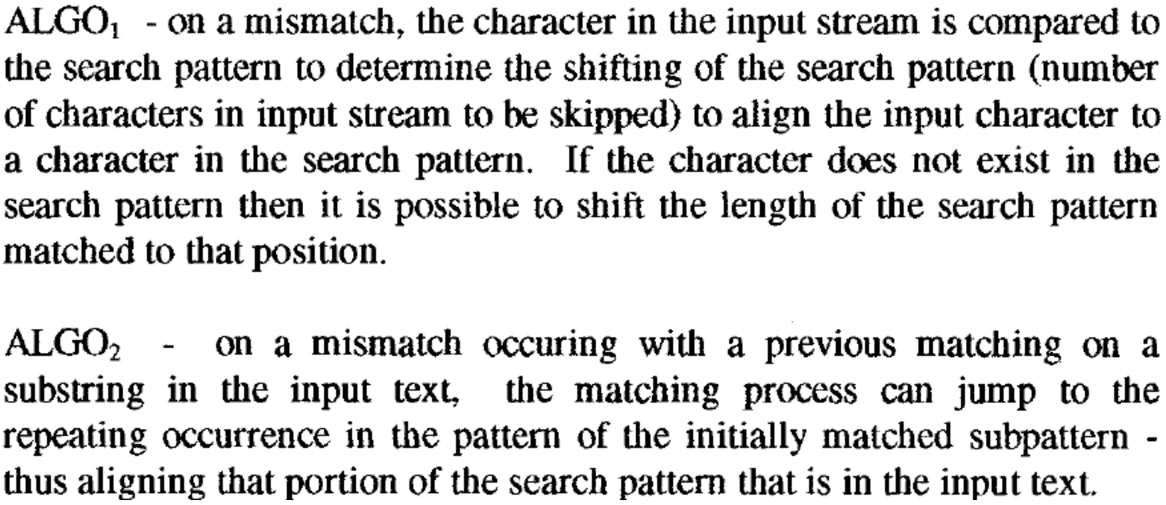
mismatch in position 13, longest repeating pattern is "a b c a" thus skip 3

P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
S									a	b	c	a	b	c	a	b
I	b	a	b	c	b	a	b	c	a	b	c	a	a	b	c	a

alignment after last shift

Figure 9.4 Example of Knuth-Morris-Pratt Algorithm

- ✓ Boyer-Moore recognized that the string algorithm could be significantly enhanced if the comparison process starts at the end of the search pattern, processing right to left versus the start of the search pattern.
- ✓ The advantage is that large jumps are possible when the mismatched character in the input stream does not exist in the search pattern which occurs frequently. This leads to two possible sources of determining how many input characters to be jumped.
- ✓ As in the Knuth-Morris-Pratt technique, any characters that have been matched in the search pattern require an alignment with that substring.
- ✓ Additionally, the character in the input stream that was mismatched also requires alignment with its next occurrence in the search pattern or the complete pattern can be moved. This can be defined as:



ALGO_1 - on a mismatch, the character in the input stream is compared to the search pattern to determine the shifting of the search pattern (number of characters in input stream to be skipped) to align the input character to a character in the search pattern. If the character does not exist in the search pattern then it is possible to shift the length of the search pattern matched to that position.

ALGO_2 - on a mismatch occuring with a previous matching on a substring in the input text, the matching process can jump to the repeating occurrence in the pattern of the initially matched subpattern - thus aligning that portion of the search pattern that is in the input text.

HARDWARE TEXT SEARCH SYSTEMS

- ✓ Software text search is applicable to many circumstances but has encountered restrictions on the ability to handle many search terms simultaneously against the same text and limits due to I/O speeds.
- ✓ One approach that off loaded the resource intensive searching from the main processors was to have a specialized hardware machine to perform the searches and pass the results to the main computer which supported the

user interface and retrieval of hits. Since the searcher is hardware based, scalability is achieved by increasing the number hardware search devices.

- ✓ The only limit on speed is the time it takes to flow the text off secondary storage (i.e., disk drives) to the searchers.
- ✓ By having one search machine per disk, the maximum time it takes to search a database of any size is the time to search one disk.
- ✓ In some systems, the disks were formatted to optimize the data flow off of the drives. Another major advantage of using a hardware text search unit is in the elimination of the index that represents the document database.
- ✓ Typically the indexes are 70 per cent the size of the actual items.
- ✓ Other advantages are that new items can be searched as soon as received by the system rather than waiting for the index to be created and the search speed is deterministic.
- ✓ When the term comparator is implemented with parallel comparators, each term in the query is assigned to an individual comparison element and input data are serially streamed into the detector.
- ✓ When a match occurs, tile term comparator informs the external query resolver (usually in the main computer) by setting status flags.
- ✓ In some systems, some of the Boolean logic between terms is resolved in the term detector hardware (e.g., in the GESCAN machine and Fast Data Finder) instead of using specially designed comparators.
- ✓ Specialized hardware that interfaces with computers and is used to search secondary storage devices was developed from the early 1970s with the most recent product being the Parasel Searcher (previously the Fast Data Finder).
- ✓ The need for this hardware was driven by the limits in computer resources.
- ✓ The typical hardware configuration is shown in Figure 9.9 in the dashed box.
- ✓ The speed of search is then based on the speed of the I/O.

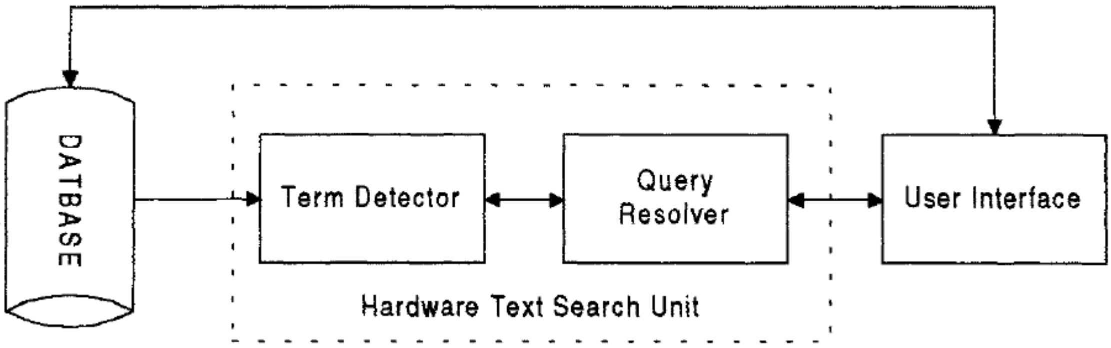


Figure 9.9 Hardware Text Search Unit

- ✓ The GESCAN system uses a text array processor (TAP) that simultaneously matches many terms and conditions against a given text stream.
- ✓ The TAP receives the query information from the user's computer and directly accesses the textual data from secondary storage.
- ✓ The TAP consists of a large cache memory and an array of tour to 128 query processors.
- ✓ The text is loaded into the cache and searched by the query processors (Figure 9.10).
- ✓ Each query processor is independent and can be loaded at any time. A complete query is handled by each query processor.
- ✓ Queries support exact term matches, fixed length don't cares, variable length "don't cares," terms restricted to specified zones, Boolean logic, and proximity.

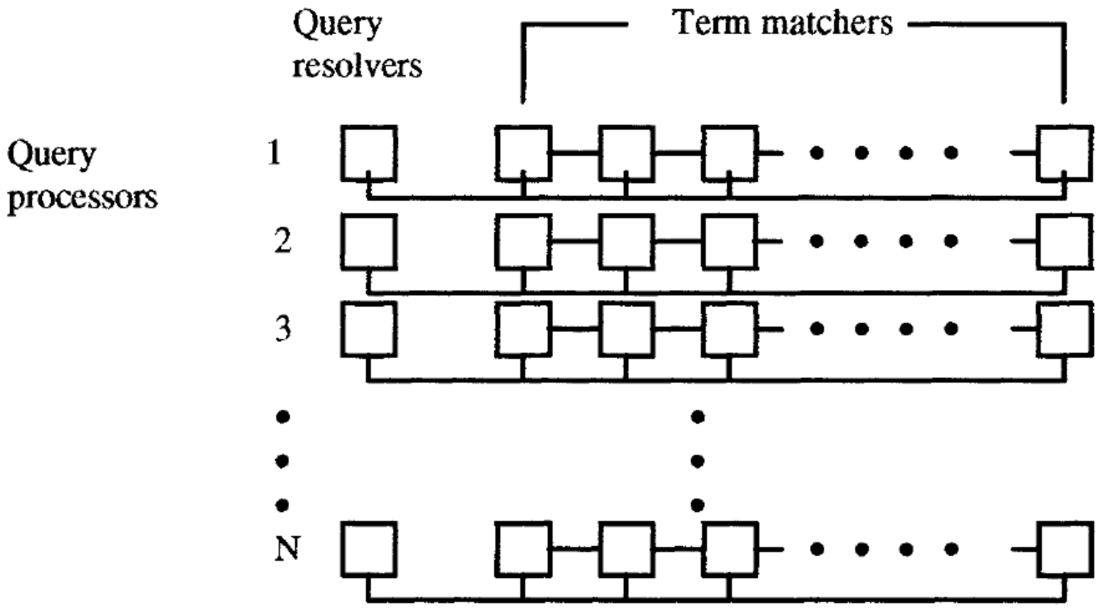


Figure 9.10 GESCAN Text Array Processor

- ✓ An example of a Fast Data Finder system is shown in Figure 9.11.
- ✓ A cell is composed of both a register cell (Rs) and a comparator (Cs).
- ✓ The input from the Document database is controlled and buffered by the microprocessor/memory and feed through the comparators. The search characters are stored in the registers.
- ✓ The connection between the registers reflects the control lines that are also passing state information.

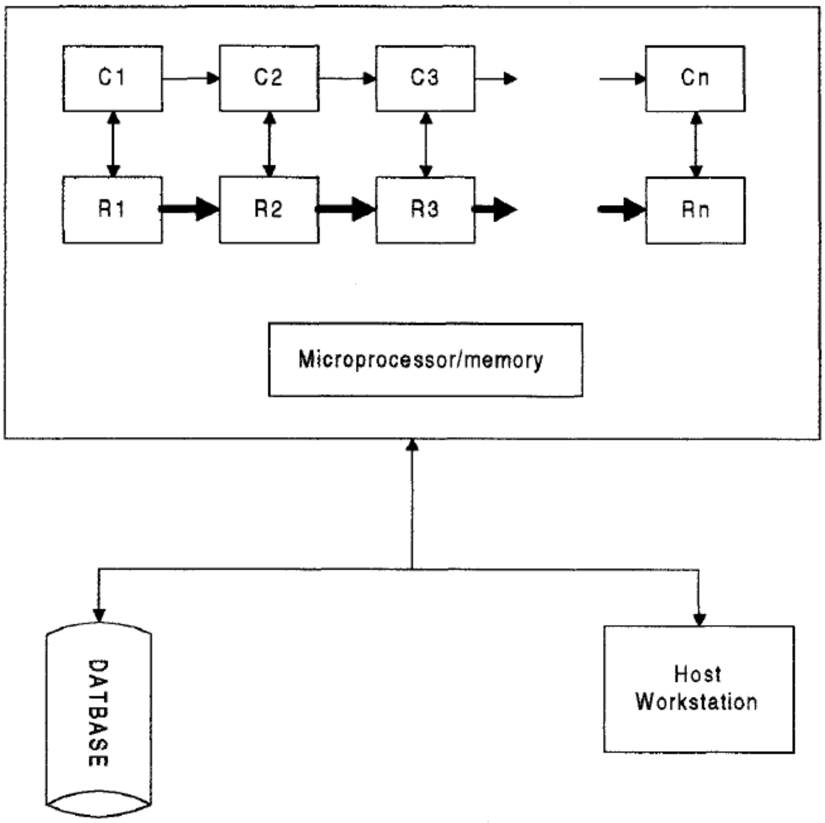


Figure 9.11 Fast Data Finder Architecture

******

PREPARED BY

RIYAZ MOHAMMED