Information Retrieval Systems

Prepared By
G Sunil Kumar
Asst.Professor

UNIT 1

Introduction to Information retrieval Systems

Contents

- **Introduction**
- Definition of Information Retrieval System
- Objectives of Information Retrieval System
- Functional Overview
- Relationship to RDBMS
- Digital Libraries and Data Warehouses
- Summary

Introduction

- Here we will define and differentiate the differences between Information Retrieval and DBMS.
- The importance of differences lies in the inability of a DBMS to provide the functions needed to process "information".
- An information system containing structured data also suffers major functional deficiencies.

Contents

- Introduction
- **Definition of Information Retrieval System**
- Objectives of Information Retrieval System
- Functional Overview
- Relationship to RDBMS
- Digital Libraries and Data Warehouses
- Summary

Definition of IRS

Def:

- An information Retrieval System is a system that is capable of storage, retrieval and maintenance of information.
  - Information here can be composed of text (including numeric and date data), images, audio, video and other multimedia objects.

- An information Retrieval System consists of a software program that facilitates a user in finding the information the user needs.

Definition of IRS

- The gauge of success of an information system is how well it can minimize the overhead for a user to find the needed information

  - **Overhead:** It is the time required to find the information needed, excluding the time for actually reading the relevant data.
  - **Aspects:** Search composition, search execution and reading non-relevant items

Contents

- Introduction
- Definition of Information Retrieval System
- **Objectives of Information Retrieval System**
- Functional Overview
- Relationship to RDBMS
- Digital Libraries and Data Warehouses
- Summary

Objectives of IRS

- **Objective:**

  To minimize the overhead of a user locating needed information.

  - **Overhead** is the time a user spends in all of the steps leading to reading an item containing the needed information. Eg: Query generation, query execution etc.

- The success of an information system is very subjective, based upon what information is needed and the willingness of user to accept overhead.

Objectives of IRS

- Measures:
  - Precision
  - Recall

$$\text{Precision} = \frac{\text{Number\_Retrieved\_Relevant}}{\text{Number\_Total\_Retrieved}}$$

$$\text{Recall} = \frac{\text{Number\_Retrieved\_Relevant}}{\text{Number\_Possible\_Relevant}}$$

Objectives of IRS

Where

Number_Possible_Relevant is the number of relevant items in the database,

Number_Total_Retrieved is the total number of items retrieved from the query

Number_Retrieved_Relevant is the number of items retrieved that are relevant to the user's search need

Objectives of IRS

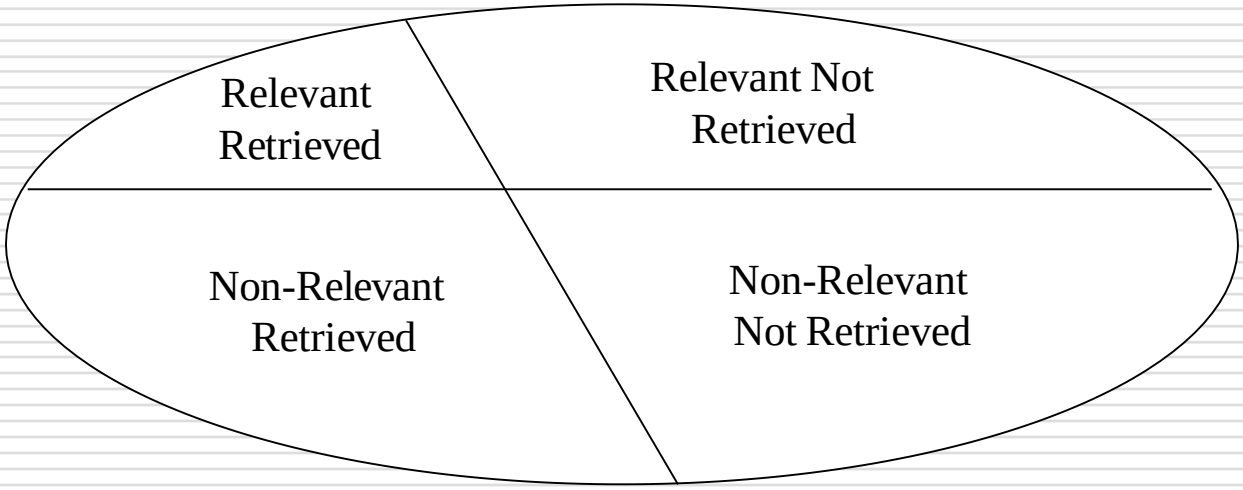


Figure1: Effects of Search on Total Document Space

Objectives of IRS

- Precision measures one aspect of information retrieval overhead for a user associated with a particular search.

- Recall gauges how well a system processing a particular query is able to retrieve the relevant items that the user is interested in seeing.

- Recall is a very useful concept, but due to the denominator (in formula on recall) is non-calculable in operational systems

Objectives of IRS

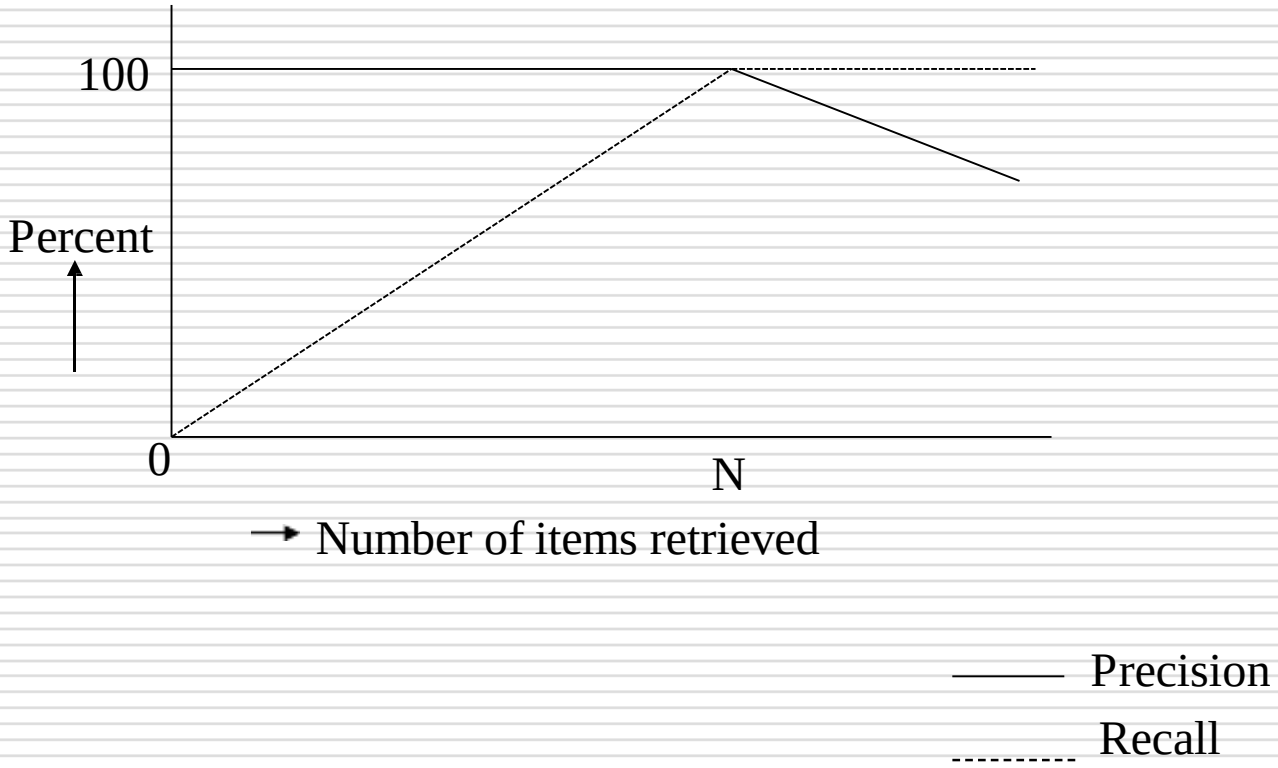


Fig: Ideal Precision and Recall

Objectives of IRS

- Precision starts off at 100 percent and maintains that value as long as relevant items are retrieved.
  - Precision is directly affected by retrieval of non-relevant items and drops close to zero.
- Recall starts off close to zero and increases as long as relevant items are retrieved.
  - Recall is not effected by retrieval of non-relevant items and hence remains at 100 percent.

Objectives of IRS

Fig: Ideal Precision/Recall graph

Here every item retrieved is relevant. Thus Precision stays at 100 percent(1.0). Recall continues to increase by moving to the right on the X-axis until it also reaches to 100 percent.

Objectives of IRS

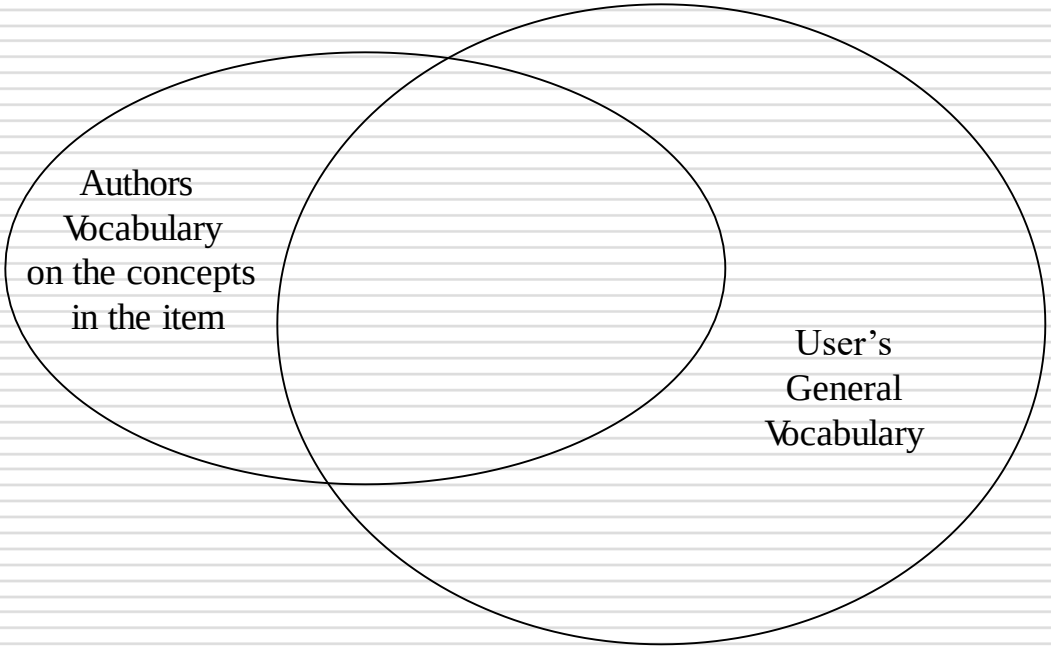


Fig: Vocabulary Domains

Contents

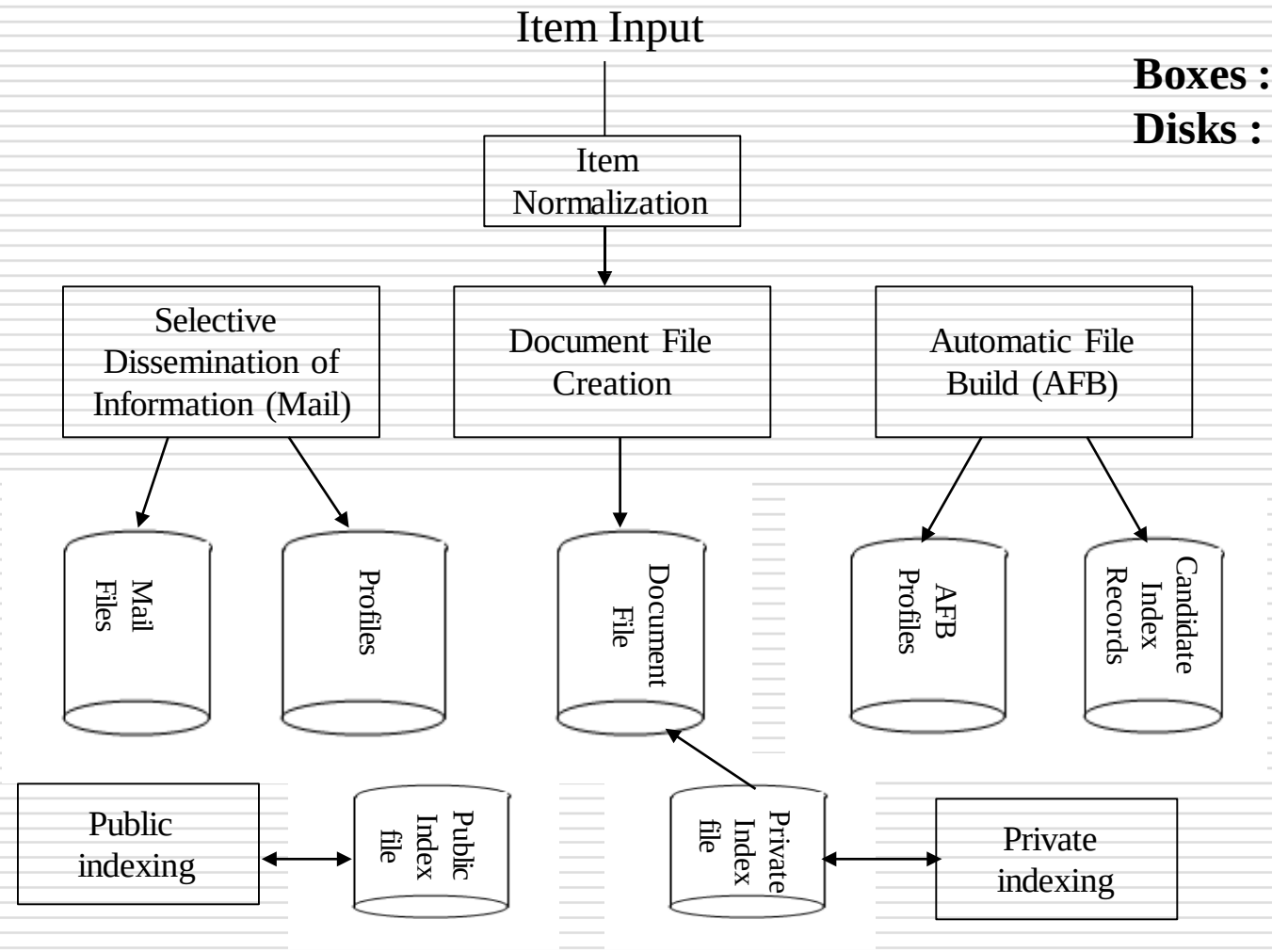
- Introduction
- Definition of Information Retrieval System
- Objectives of Information Retrieval System
- **Functional Overview**
- Relationship to RDBMS
- Digital Libraries and Data Warehouses
- Summary

Functional Overview

- The total information storage and retrieval system consists of four major functional processes:
  - Item Normalization
  - Selective dissemination of information (i.e. Mail)
  - Archival Document Database Search
  - Index database search along with Automatic File Build Process

The next figure shows the logical view of these capabilities in a single integrated information retrieval system.

Fig: Total Information Retrieval System



Boxes : Functions
Disks : Data storage

Functional Overview

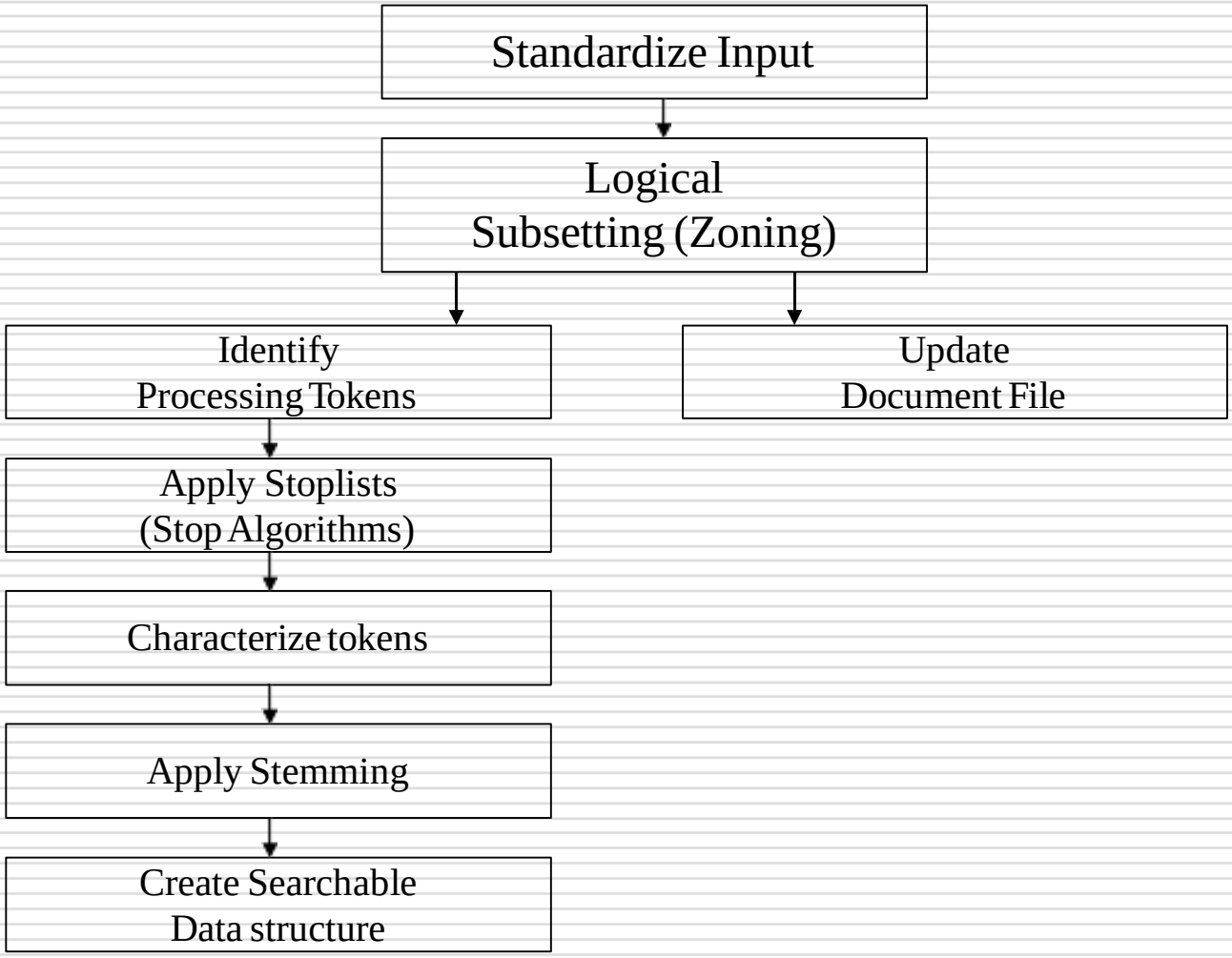
1. Item Normalization

- Normalize the incoming items to a standard format
- Provides logical restructuring of the item.
- Additional operations are needed to create a searchable data structure:
  - Identification of Processing tokens
  - Characterization of tokens
  - Stemming (eg: removing word endings) of tokens.

The processing tokens and their characterization are used to define the searchable text from the total received text.

The following figure shows the normalization process.

Fig: Text Normalization Process



Functional Overview

- Standardizing the input takes the different external formats of input data and performs the translation to the formats acceptable to the system (eg: translation of foreign languages into Unicode)
- One standard encoding that covers English, French, Spanish, etc. is ISO-Latin
- Multimedia adds an extra dimension to the normalization Process.
- If the input is video the likely digital standards will be: MPEG-2, MPEG-1 etc.
- MPEG (Motion Picture Expert Group) are the most universal standards for higher quality video.

Functional Overview

- ***Zoning:*** It is the process to parse the item into logical subdivisions that have meaning to the user
  - Used to increase the precision of a search and optimize the display
- Identify the Processing tokens
  - Consists of determining a word
  - Systems determine words by dividing input symbols into three classes:
    - Valid word Symbols
    - Inter-word symbols
    - Special processing symbols
  - ***Word:*** It is defined as a contiguous set of word symbols bounded by inter-word symbols (eg: of word symbols are Alphabetic characters and numbers, eg of inter-word symbols are: blanks, periods and semicolons)

Functional Overview

- Stop List Algorithm is applied to the list of potential processing tokens.
  - ***Objective of Stop function:*** To save system resources by eliminating from the set of searchable processing tokens those that have little value to the system.
  - Stop lists are commonly found in most systems and consists of words (Processing tokens) whose frequency and/or semantic use make them of no value as a searchable token.
  - The rank frequency law of Ziph
    - **Frequency * Rank = Constant**

      where Frequency = no. of times a word occurs and

      Rank = rank order of the word

Functional Overview

- The next step in finalizing on processing tokens is identification of any specific word characteristics.
- Once the potential processing token has been identified and characterized, most systems apply stemming algorithms to normalize the token to a standard semantic representation
- The decision to perform stemming is a trade-off between precision of a search Vs standardization to reduce system overhead in expanding a search term to similar token representations.
- Once the processing tokens have been finalized based upon the stemming algorithm, they are used as updates to the searchable data structure.

Functional Overview

2. Selective Dissemination of Information (Mail)

- This process provides the 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 and user mail files.

Functional Overview

3. Document database Search

- This process provides the capability for a query to search against all items received by the system.
- This process is composed of the search process, user entered queries and Document database which contains all items that have been received, processed and stored by the system.
- 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 not edited) once received.

Functional Overview

4. Index Database Search

- A user may want to save the interested item for future reference. This is accomplished via the Index 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.
- A good analogy to an index file is the card catalog in a library.
- The index database search Process provides the capability tom create indexes and search them.

Functional Overview

- 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 Process is called a **Combined File Search.**
- In an ideal system the index record could reference portions of items versus the total item

Functional Overview

- Two classes of Index files: Public and Private
  - Every user can have one or more Private Index Files leading to a very large number of files
  - Private Index File
    - References only a small subset of the total number of items in the Document Database.
    - Typically have very limited access lists.
  - Public Index Files
    - Maintained by professional library services personnel and typically index every item in the Document Database.
    - Have access lists that allow any one to search and retrieve data.

Functional Overview

- To assist the users in generating indexes, especially the professional indexers, the system provides a process called Automatic File Build.

- The capability to create Private and Public Index Files is frequently implemented via a Structured DBMS.

Functional Overview

- From a system perspective, the multimedia data is not logically its own data structure.
- It will reside almost entirely in the area described as the Document database
- The correlation between the multimedia and the textual domains will be either via Time or Positional synchronization
  - Time synchronization is an ex. of transcribed text from audio or composite video sources.
  - Positional synchronization is where the multimedia is localized by a hyperlink in a textual item.

Contents

- Introduction
- Definition of Information Retrieval System
- Objectives of Information Retrieval System
- Functional Overview
- **Relationship to DBMS**
- Digital Libraries and Data Warehouses
- Summary

Relationship to DBMS

- Information Retrieval System is software that has the features and functions require to manipulate " information" items
- A DBMS is optimized to handle structured data.
  - Structured data is well defined data represented by tables.
- 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

Relationship to DBMS

- The integration of DBMS and Information Retrieval systems is very important.
- One of the first commercial databases to integrate the two systems into a single view is the INQUIRE DBMS (past 15 years)
- A more current example is the ORACLE DBMS that offers an imbedded capability called CONVECTIS
  - It is Information Retrieval system that uses a comprehensive thesaurus which provides the basis to generate “themes” for a particular item

Relationship to DBMS

- The INFORMIX DBMS has the ability to link to RetrievalWare to provide integration of structured data and Information along with functions associated with Information retrieval systems.

Contents

- Introduction
- Definition of Information Retrieval System
- Objectives of Information Retrieval System
- Functional Overview
- Relationship to DBMS
- **Digital Libraries and Data Warehouses**
- Summary

Digital Libraries and DWHs

- There is significant overlap between these two systems and information storage and retrieval systems.
- Digital libraries, DWHs and Information retrieval systems are the repositories of information.
  - ***Goal:*** to satisfy user information needs

Contents

- Introduction
- Definition of Information Retrieval System
- Objectives of Information Retrieval System
- Functional Overview
- Relationship to DBMS
- Digital Libraries and Data Warehouses
- **Summary**

Summary

- This unit places into perspective a total Information Storage and Retrieval System. This perspective introduces new challenges to the problems that need to be theoretically addressed and commercially implemented.

- From a theoretical perspective, efficient scalability of algorithms to systems with gigabytes and terabytes of data, operating with minimal user search statement information and making maximum use of all functional aspects of an information system need to be considered.

Summary

- The dissemination systems or mail files to modify ranking algorithms and combining the search of structured information fields and free text into a consolidated weighted output are examples of potential new areas of investigation.
- Understanding the differences between Digital Libraries and Information Retrieval systems will add an additional dimension to the potential future development of systems.
- The collaborative aspects of digital libraries can be viewed as a new source of information that dynamically could interact with information retrieval techniques.

Information Retrieval System Capabilities

Contents

- **Search Capabilities**
- Browse Capabilities
- Miscellaneous Capabilities
- Standards
- Summary

1. Search Capabilities

- The search capabilities address both Boolean and Natural Language Queries
- The algorithms used for searching are called Boolean, natural language processing and probabilistic.
- Probabilistic algorithms use frequency of occurrence of processing tokens in determining similarities between queries and items.
- The systems such as TOPIC, RetrievalWare and INQUERY allow for natural language queries.

Continued…

- **Objective:**
  - To allow for a mapping between a user's specified need and items in the information database that will answer that need.

1.1 Boolean Logic

- It allows a user to logically relate multiple concepts together to define what information is needed.
- Boolean functions apply to processing tokens identified anywhere within an item.
- Typical Boolean operators are AND, OR and NOT, and are implemented using intersection, set union and set difference procedures

Continued…

- **Ex:** Find any item containing any two of the following terms: ‘AA’, ‘BB’, ‘CC’. This can be expanded into a Boolean search that performs an AND between all combinations of two terms and ‘OR’s that results together ((AA AND BB) or (AA AND CC) or (BB AND CC))

- Most information retrieval systems allow Boolean operations and natural language interfaces

Fig: Use of Boolean Operators

SEARCH STATEMENT	SYSTEM OPEARTION
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

Continued…

- 1.2 Proximity
  - It 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.
  - It is used to increase the precision of a search.
  - The typical format for proximity is:

    TERM 1 within “m” “units” of TERM 2

    The distance operator “m” is an integer number

    “units” are in Characters, Words, Sentences, or Paragraphs

Continued…

- The proximity relationship contains a direction operator indicating the direction (before or after) that the second term must be found within the number of units specified. Default is either direction.
- A special case of Proximity operator is the Adjacent (ADJ) operator that normally has a distance operator.
- Another special case is where the distance is set to zero (within the same semantic unit)

Fig: Use of Proximity

SEARCH STATEMENT	SYSTEM OPEARTION
"Venetian" ADJ "Blind"	Would find items that mention a Venetian Blind on a window but not items discussing a Blind Venetain
"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.

Continued…

- 1.3 Contiguous Word Phrases (CWP)
  - A CWP is both a way of specifying a query term and special search operator.
  - A CWP is two or more words that are treated as a single semantic unit.
  - **EX:** "United States of America"
    - It is four words that specify a search term representing a single specific semantic concept (a country)
  - A CWP also acts like a special search operator that is similar to the proximity operator but allows for additional specificity

Continued…

- If two terms are specified, CWP and Proximity operator are identical.
- For CWP with more than two terms the only way of creating an equivalent search statement using proximity and Boolean operators is via nested Adjacencies.
- A CWPs are called Literal Strings in WAIS (Wide Area Information Servers) and Exact Phrases in RetrievalWare.
- In WAIS multiple Adjacency (ADJ) operators are used to define a Literal String. **Ex:** "United" ADJ "States" ADJ "of" ADJ "America"

Continued…

- 1.4 Fuzzy Searches
  - Fuzzy searches provide the capability to locate spelling 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 to words in the database.

Continued…

- **EX:** Term entered is “computer”

  Fuzzy search would automatically include the following words from the information database: “computer”, “compiter”, “conputer”, “computter”, “compute”.

- Fuzzy searching has its maximum utilization in systems that accept items that have been Optical Character Read (OCR)
- In the OCR process a hardcopy item is scanned into a binary image (usually at a resolution of 300 dots per inch or more)
- The OCR process is a pattern recognition process that segments the scanned image into meaningful subregions.
- The OCR process will then determine the character and translate it to an internal computer encoding (ex: ASCII or other)

Continued…

- 1.5 Term Masking

  - It 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.
  - Two types of search term masking are: fixed length
    variable length

Continued…

- Fixed length term masking is a single position mask. It masks out any symbol in a particular position or lack of that position in a word.
- 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.

“* COMPUTER”	Suffix Search
“COMPUTER*”	Prefix Search
“*COMPUTER*”	Imbedded String Search

Fig: Term masking

SEARCH STATEMENT	SYSTEM OPEARTION
Multi$national	Matches "multi-national," "multiynational," "multinational". Does not match "multi national" single is two processing tokens.
computer	Matches , "minicomputer" "microcomputer" or "computer"
comput*	Matches "computers" "computing", "computes"
comput	Matches "microcomputers", "minicomputing", "compute"

Continued…

- 1.6 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.
  - A user could enter inclusive (e.g., “125-425” or “4/2/93-5/2/95” for numbers and dates) to infinite ranges (“>125,” “<=233,” representing “Greater Than” or “Less Than or Equal”) as part of query.

Continued…

- 1.7 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.
  - Concept class is a tree structure that expands each meaning of a word into potential concepts that are related to the initial term.

Fig: Thesaurus for term “computer”

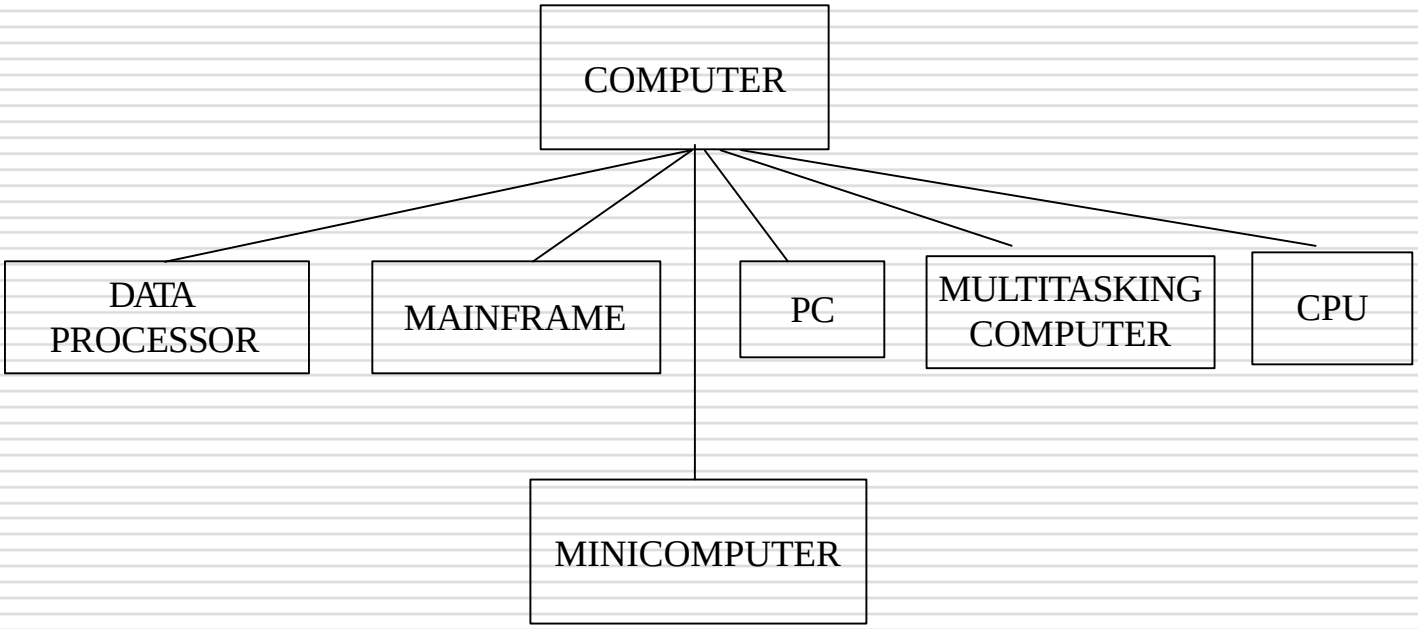
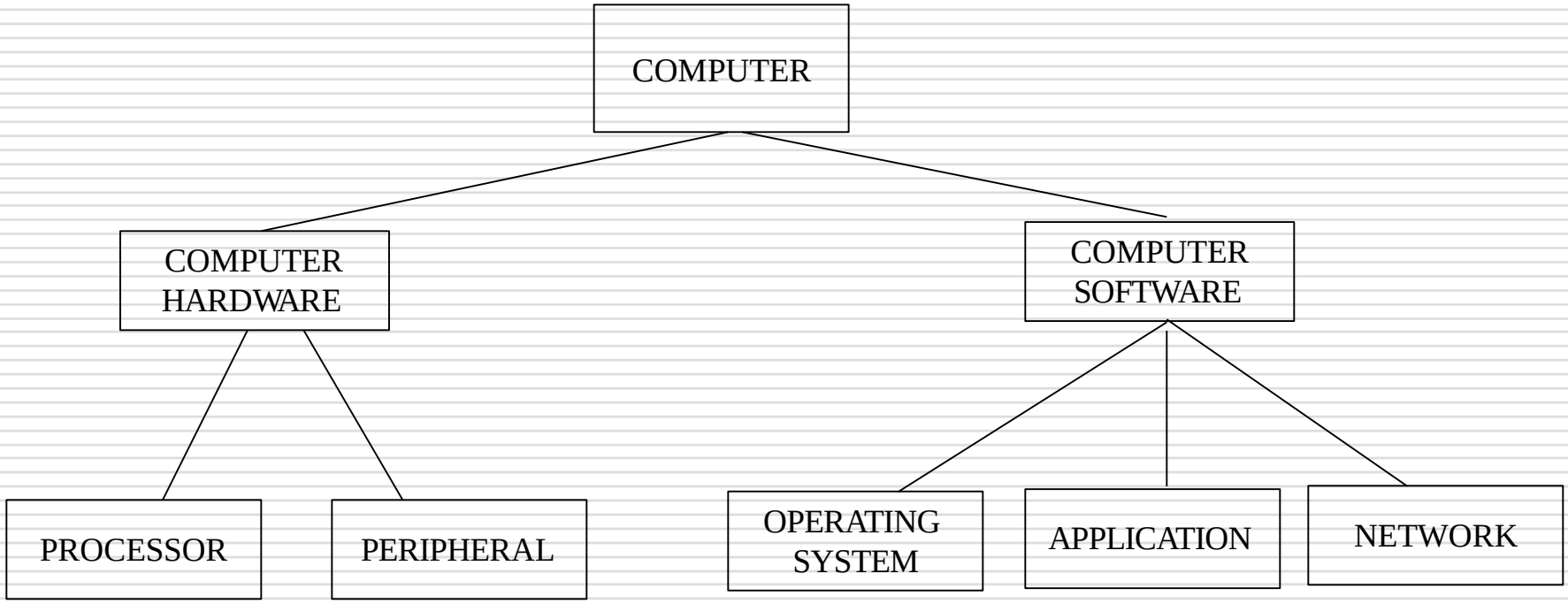


Fig: Hierarchical concept Class Structure for "computer"



Continued…

- 1.8 Natural Language Queries

  - Natural Language Queries allow a user to enter a prose statement that describes the information that the user wants to find.
  - The longer the prose, the more accurate the results returned.
  - The most difficult logic case associated with Natural language queries is the ability to specify negation in the search statement and have the system recognize it as negation.

Continued…

- To accommodate the negation function and provide users with a transition to the natural language systems, most commercial systems have a user interface that provides both a natural language and Boolean logic capability.
- Negation is handled by the Boolean portion of a search.
- Natural language interfaces improve the recall of systems with a decrease in precision when negation is required.

Continued…

- 1.9 Multimedia Queries
  - The user interface becomes far more complex with the introduction of the availability of multimedia items.
  - The current systems only focus on specification of still images.
  - Audio sources are converted to searchable text via audio transcription. This allows queries t be applied to the text.
  - But, like OCR output, the transcribed audio will contain many errors.
  - Thus, the search algorithms must allow for errors in the data. The errors are very different compared to OCR.

Continued…

- OCR errors will usually create a text string that is not a valid word..
- In ASR (Automatic Speech Recognition), all errors are other valid words since ASR selects entries ONLY from dictionary of words.
- Audio also allows the user to search on specific speakers, since speaker identification is relatively accurate against audio sources.
- The correlation between different parts of a query against different modalities is usually based upon time or location. Most common is Time.

Contents

- Search Capabilities
- **Browse Capabilities**
- Miscellaneous Capabilities
- Standards
- Summary

2. Browse Capabilities

- 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

  Data visualization
- From these summary displays, the user can select the specific items and zones within the items for display

Continued….

- The system also allows for easy transitioning between the summary displays and review of specific items.
- If searches resulted in high precision, then the importance of 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 the need.

2.1 Ranking

- With the introduction of ranking based upon predicted relevance values, the status summary displays the relevance score.
  - **Relevance score:** It is an estimate of the search system on how closely the item satisfies the search statement.
    - 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.

2.1 Ranking

- Theoretically every item in the system could be returned but many of the items will have a relevance value of 0.0.
- 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.
- In many circumstances Collaborative Filtering is providing an option for selecting and ordering output
  - In this, users when reviewing items provide feedback to the system on the relative value of the item being accessed.

2.1 Ranking

- The system accumulates the various user rankings and uses this information to order the output for other user queries that are similar.
- Collaborative filtering has been very successful in sites such as AMAZON.COM, MovieFinder.com and CDNow.com in deciding what products to display to users based upon their queries.

- Information visualization is also being used in displaying individual items and the terms that contributed to the item's selection.

2.2 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.

2.2 Zoning

- Related to zoning for use in minimizing what an end user needs to review from a hit item is the idea of locality and passage based search and retrieval.
- Here the basic search unit is not complete item, but an algorithmic defined subdivision of the item.
- This is known as passage retrieval where the item is divided into uniform sized passages that are indexed and locality based retrieval where passage boundaries can be dynamic.

2.3 Highlighting

- The indication, frequently highlighting, lets the user quickly focus on the potentially relevant parts of the text to scan for item relevance.
- It 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.
- Information visualization appears to be a better display process to assist in helping the user formulate the query than highlighting items.

Contents

- Search Capabilities
- Browse Capabilities
- **Miscellaneous Capabilities**
- Standards
- Summary

3. 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.
  - Vocabulary Browse
  - Iterative searching and search history log
  - Canned queries

3.1 Vocabulary browse

- Vocabulary Browse provides knowledge on the processing tokens available in the searchable database.
- It provides the capability to display alphabetical sorted order words from the document database.
- Logically, all unique words (processing tokens) in the database are kept in sorted order along with the 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 the dictionary around the entered text.

3.1 Vocabulary browse

Below table shows what is seen in vocabulary browse if the user enters **“comput”**

TERM	OCCURRENCES
Computation	265
Comput	1245
Computen	1
Computer	10,800
Computerize	18
Computes	29

Fig: Vocabulary Browse list with entered term “comput”

3.1 Vocabulary browse

- The system indicates what word fragment the user entered and then alphabetically displays other words found in the database.
- 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.

3.2 Iterative 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 the relevant items is called iterative search.

3.2 Iterative search and search history log

- During a login session, a user could execute many queries to locate the needed information.
- To facilitate locating previous searches as starting points for new searches, search history logs are available.
- The search history log is the capability to display all the previous searches that were executed during the current session.

3.3 Canned Query

- The capability to name a query and store it to be retrieved and executed during a later user sessions is called canned or stored queries.
- A canned allows a user to create and refine a search that focuses on the user's general area of interest one time and then retrieve it to add additional search criteria to retrieve data that is currently needed.
- Queries that start with a canned query are significantly larger than ad hoc queries.

3.4 Multimedia

- To display more aggregate data, textual interfaces sometimes allow for clustering of the hits and then use of graphical display to show a higher level view of the information.

UNIT II

Cataloging and Indexing

Introduction

- ***Indexing:***
  - The transformation from the received item to the searchable data structure is called Indexing.
  - This process can be manual or automatic
  - Creates the basis for direct search of items in the Document Database or indirect search via Index Files.
- Information extraction is closely associated with the indexing process.
  - ***Goal:*** To extract specific information to be normalized and entered into a structured database (DBMS)

Introduction

- Information extraction differs because it focuses on very specific concepts and contains a transformation process that modifies the extracted information into a form compatible with the end structured database. This process is referred as Automatic file Build.

Contents

- **History of Indexing**
- Objectives of Indexing
- Indexing Process
- Automatic indexing
- Information Extraction
- Summary

1. History of Indexing

- Indexing is the oldest technique for identifying the contents of items to assist in their retrieval.
- MARC (MAchine Readable Cataloging)
  - Standardizes the structure, contents and coding of bibliographic records.
- Objective of Cataloging:
  - To give access points to a collection that are expected and most useful to the users of the information.
- The earliest commercial cataloging system is DIALOG
  - Developed by Lockheed Corporation in 1965 for NASA

Contents

- History of Indexing
- **Objectives of Indexing**
- Indexing Process
- Automatic indexing
- Information Extraction
- Summary

2. Objectives of Indexing

- The full text searchable data structure for items in the Document File provides a new class of indexing called total document indexing.

- The availability of items in electronic form changes the objectives of manual indexing. The source information (frequently called citation data) can automatically be extracted.

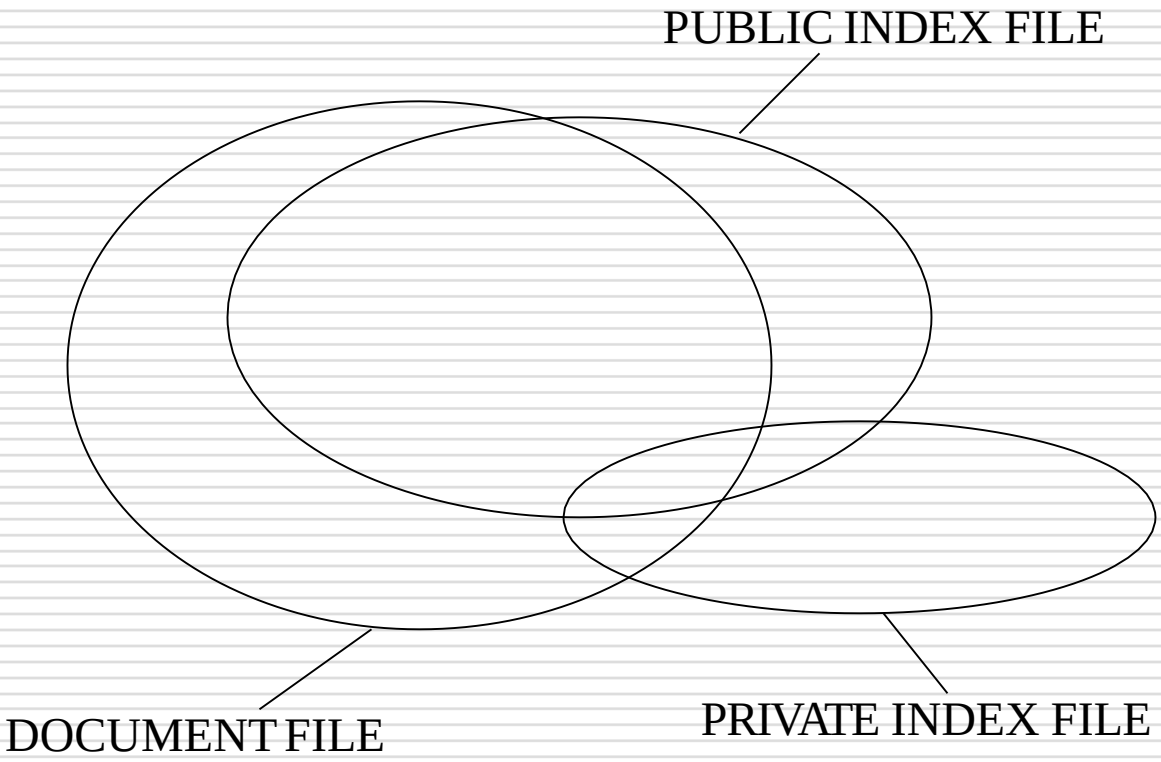
Contents

- History of Indexing
- Objectives of Indexing
- **Indexing Process**
- Automatic indexing
- Information Extraction
- Summary

Indexing process

- When an organization with multiple indexers decides to create a public or private index some procedural decisions are required.
  - The scope of indexing to define what level of detail the subject index will contain. This is based upon usage scenarios of the end users.
  - The need to link index terms together in a single index for a particular concept.

Fig: Items Overlap between Full Item Indexing, Public File Indexing and Private File Indexing



Continued…

- **1. Scope of Indexing:** There are two factors involved in deciding on what level to index the concepts in an item.
  - Exhaustivity
  - Specificity

Exhaustivity: It is the extent to which the different concepts in the item are indexed.

Specificity: It relates to the preciseness of index terms used in indexing

Continued…

- Another decision on indexing is what portions of an item should be indexed. Simplest case is to limit the indexing to the Title or Title and Abstract Zones.
- **Weighting:** it is the process of assigning an importance to an index term's use in an item. It is not common in manual indexing systems.
- Weight should represent the degree to which the concept associated with the index term is represented in the item
- The manual process of assigning weights adds additional overhead and requires more complex data structure to store the weights.

Continued…

- **2. Precoordination and Linkages:** 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 precoordination.
- When index terms are not coordinated at index time, the coordination occurs at search time. This is called post coordination. i.e., coordinating terms after the indexing process.
- Post coordination is implemented by "AND" ing index terms together
- Factors that must be determined in the linkage process are the number of terms that can be related, any ordering constraints on the linked terms and any additional descriptors are associated with index terms.

Fig: Linkage of Index Terms

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

Contents

- History of Indexing
- Objectives of Indexing
- Indexing Process
- **Automatic indexing**
- Information Extraction
- Summary

4. Automatic Indexing

- Automatic Indexing is the capability for the system to automatically determine the index terms to be assigned to an item.
- The simplex 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.
- Automatic indexing requires only a few seconds or less of computer time based upon the size of processor and complexity of algorithms to generate index.
- **Adv of Human Indexing:** The ability to determine concept abstraction and judge the value of a concept.

Continued….

- **Dis Adv of Human Indexing over Automatic Indexing:** Cost, Processing time and consistency.
- Processingtime 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.
- **Adv of Automatic Indexing:** The predictability of algorithms. If the indexing is being performed automatically , by an algorithm, there is consistency in the index term selection process.

Continued….

- Indexes resulting from automated indexing fall into two classes:
  - Weighted
  - Unweighted
- **Unweighted indexing system:**
  - The existence of an index term in a document and sometimes its word location(s) are kept as part the searchable data structure.
  - No attempt is made to discriminate between the value of index terms in representing concepts in the item.
- **Weighted indexing system:**
  - An attempt is made to place a value on the index term's representation of its associated concept in the document.
  - An index term's weight is based upon a function associated with the frequency of occurrence of the term in the item

4.1 Indexing by Term

- There are two major techniques for creation of the index.
  - Statistical
  - Natural language
- **Statistical techniques:**
  - These 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 databases.
  - Can be based upon vector models and probabilistic models with a special case being Bayesian models.

Continued…

- **Bayesian models:**
- This approach could be applied as part of index term weighting, but usually is applied as part of retrieval process by calculating relationship between an item and specific query.
- A Bayesian network is a directed acyclic graph in which each node represents a random variable and the arcs between the nodes represent a probabilistic dependence between the node and its parents

Continued…

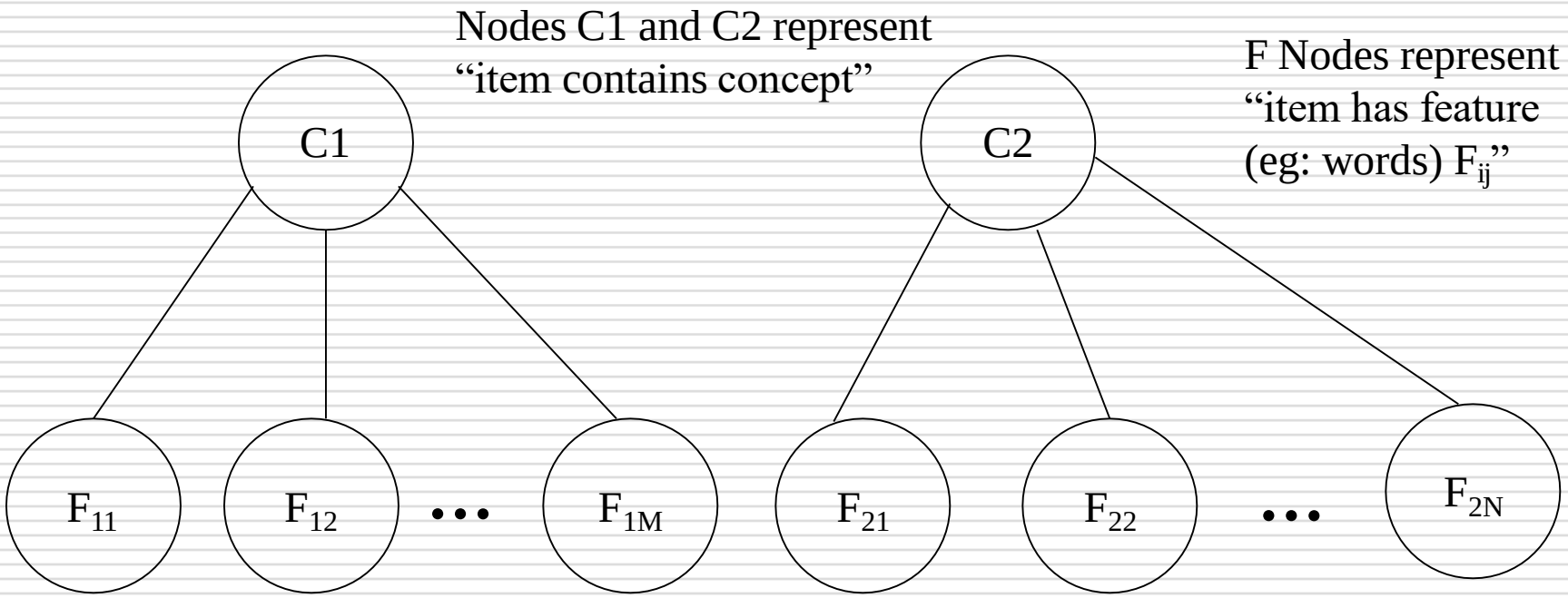


Fig: Two level Bayesian Network

Continued…

- The network could also be interpreted as C representing concepts in a query and F representing concepts in an item.
- The **goal** is to calculate the probability of C_i given F_{ij}. To perform that calculation two sets of probabilities are needed:
  - The prior probability $P(C_i)$ that an item is relevant to concept C
  - The conditional probability P(Fij/Ci) that the features Fij where j=l,m are present in an item given that the item contains topic Ci
- The automatic indexing task is to calculate the posterior probability $P(C_i/F_{il}, ….., F_{im})$, the probability that the item contains concept Ci, given the presence of features Fij.

Continued…

- The Bayes inference formula that is used is:

$P(Ci/F_{il}, ….., F_{im}) = P(Ci)\ P(F_{il}, ….., Fim/Ci)\backslash\ P(F_{il}, ….., F_{im})$.

- If the goal is to provide ranking as the result of a search by the posteriors, the Bayes rule can be simplified to a linear decision rule:

$g(Ci/F_{il}, ….., Fim) = \Sigma k\ I(F_{ik})\ w(F_{ik}, Ci)$where

$I(F_{ik})$ is an indicator variable that equals 1 only

if F_{ik} is present in the item (equals zero otherwise)

w is coefficient corresponding to a specific

feature/concept pair.

Continued…

- A careful choice of w produces a ranking in decreasing order that is equivalent to the order produced by the posterior probabilities.
- **Interpreting** the coefficients, **w**, as weights corresponding to each feature (eg: index term) and the function **g** as the sum of the weights of the features, the **result** of applying the formula is a **set of term weights**.

Continued…

- Natural Language Processing:
  - The DR-LINK (Document Retrieval through LINguistic Knowledge) system processes items at the morphological, lexical, semantic, syntactic and discourse levels.
  - Each level uses information from the previous level to perform its additional analysis.
  - The discourse level is abstracting information beyond the sentence level and can determine abstract concepts using predefined models of event relationships.
  - Normal automatic indexing does a poor job at identifying and extracting “verbs” and relationships between objects based verbs

4.2 Indexing by Concept

- 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.
- Ex. Of system uses Concept Indexing is- MatchPlus system.
- MatchPlus system uses neural networks to facilitate machine learning of concept/word relationships and sensitivity to similarity of use.
- The system goal is to be able to determine from the corpus of items, word relationships (eg. Synonyms) and the strength of these relationships and use that information in generating context vectors.

Continued.....

- The interpretation of components for concept vectors is exactly the same as weights in neural networks.
- Two neural networks are used.
  - One neural network learning algorithm generates stem context vectors that are sensitive to similarity of use.
  - Another neural network performs query modification based upon user feedback.
- 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 ~ 0 if word k is not associated with feature j
  - V^k negative if word k contradicts feature j

4.3 Multimedia Indexing

- The first pass in most cases is a conversion from the analog input mode into a digital structure.
- Then algorithms are applied to the digital structure to extract the unit of processing of the different modalities that will be used to represent the item.
- Creation of multimedia presentations are becoming more common using Synchronized Multimedia Integration Language (SMIL)
  - It is a mark-up language designed to support multimedia presentations that integrate text with audio, images and video.
- Thus indexing must include a time-offset parameter Vs physical displacement.

Contents

- History of Indexing
- Objectives of Indexing
- Indexing Process
- Automatic indexing
- **Information Extraction**
- Summary

5. Information Extraction

- There are two processes associated with information extraction:
  - Determination of facts to go into structured fields in a database
  - 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.
  - **Goal:** to process incoming items and extract index terms that will go into a structured database.
  - This differs from indexing in that its objective is to extract specific types of information Vs understanding text of the document
- An IRS goal is to provide an in-depth representation of the total contents of an item

Continued.....

- Information extraction system only analyzes those portions of a document that potentially contain information relevant to the extraction criteria.
- The objective of data extraction is to update a structured database.
- The process is very similar to the natural language processing.

Data Structures

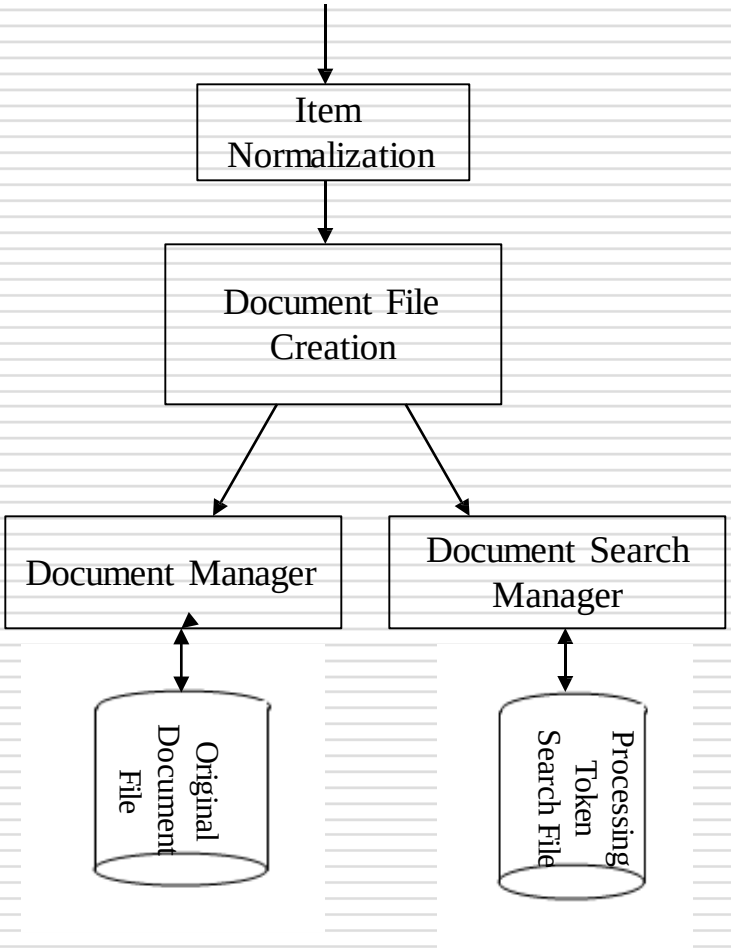
Contents

- Introduction to data structure
- Stemming Algorithms
- Inverted File Structure
- N-Gram data structure
- PAT data structure
- Signature file structures
- Hypertext and XML data structures

Introduction

- From an IRS perspective, two aspects of a data structure that are important are – its ability to represent concepts and their relationships and how well it supports location of those concepts.
- 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 "Document Manager".
  - Other data structure contains the processing tokens and associated data to support search.

Fig: Major Data Structures



Contents

- Introduction to data structure
- **Stemming Algorithms**
- **Inverted File Structure**
- **N-Gram data structure**
- **PAT data structure**
- **Signature file structures**
- **Hypertext and XML data structures**

2. Stemming Algorithms

- One of the first transformations often applied to data before placing it in searchable data structure is **stemming.**
  - It reduces the diversity of representations of a concept (word) to a canonical morphological representation.
  - **Risk with stemming:** concept discrimination information may be lost in the process, causing a decrease in precision and ability for ranking to be performed.
  - **Adv:** stemming has the potential to improve recall.

Continued…

- **Goal:** To improve performance and require less system resources by reducing the number of unique words
- A system designer can trade off the increased overhead of stemming in creating processing tokens versus reduced search time overhead of processing query terms.

Continued…

- These 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 variant to a single representation (stem).
- Stem carries the meaning of the concept associated with the word and affixes (endings) introduce slight modifications to the concept.

Continued…

- Ex: The stem “comput” could associate “Computable,computability,computation,computational, computed, computing, computer, computerize to one compressed word.
- Stemming of words “calculate, calculates, calculation, calculations, calculating” to a single stem (“calculat”) insures whichever of those terms is entered by the user.

Continued…

- In contrast, stemming cannot improve, but has the potential for decreasing precision.
  - The precision value is not based on finding all relevant items but just minimizing the retrieval of non-relevant items.
- Stemming can also cause problems for Natural Language Processing (NLP) systems by causing loss of information needed for aggregate levels of NLP.
- The most common stemming algorithm removes suffixes and prefixes, sometimes recursively , to derive final stem.

Continued…

- Techniques such as table lookup and successor stemming provide alternatives that require additional overheads.
- Table lookup requires a large data structure.
- Successor stemmers determine prefix overlap as the length of stem is increased.
  - This information can be used to determine the optimal length for each stem.

Continued…

- The affix removal techniques removes prefixes and suffixes from terms leaving the stem.
- Most stemmers are iterative and attempt to remove the longest prefixes and suffixes.
- Stemming is applied to user's query as well as to the incoming text.
- If the transformation moves the query term to a different semantic meaning, the user will not understand why a particular item is returned.

2.1 Porter Stemming Algorithm

- The porter algorithm is the most commonly accepted algorithm, but it leads to loss of precision and introduces some anomalies.

- This algorithm is based upon a set of conditions of the stem, suffix and prefix and associated actions given the condition.

Continued…

- Some examples of stem conditions are:
  - 1. The measure, m, of a stem is a function of sequences of vowels V followed by a consonant C, then m is:

    $C(VC)^mV$ 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

Continued…

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.

- Suffix conditions take the form Current suffix==pattern
- Actions are in the form old_suffix->new_suffix

2.2 Dictionary Look-Up Stemmers

- In this approach, simple stemming rules may be applied. The rules are taken from those that have the fewest exceptions (eg: removing pluralization from nouns)
- 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 RetrievalWare systems.
- The INQUERY system uses a stemming technique called Kstem

Continued….

- Kstem is a morphological analyzer that conflates word variants to a root form.
- It tries to avoid collapsing words with different meanings into the same root..
- Ex: “memorial” and “memorize” reduce to “memory”. But “memorial” and “memorize” are not synonyms and have very different meanings.
- Kstem, like other stemmers associated with Natural Language Processors and dictionaries, returns words instead of truncated word forms.

Continued….

- Kstem requires a word to be in dictionary before it reduce one word form to another.
- Some endings are always removed , even if root form is not found in dictionary(eg: ‘ness’, ‘ly’).
- If the word being processed is in the dictionary, it is assumed to be unrelated to the root after stemming and conflation is not performed(eg: ‘factorial’needs to be in the dictionary or it is stemmed to’factory’).
- It is necessary to explicitly map the word variant to the root desired(eg: ‘matrices’ to ‘matrix’)

Continued….

- Kstem system uses the following six major data files to control and limit the stemming process:
  - Dictionary of words
  - Supplemental list of words for the dictionary
  - Exceptions list for those words that should retain an ‘e’ at the end (eg: “suites” to “suite” but “suited”to “suit”)
  - Direct_conflation – allows definition of direct conflation via word pairs that override the stemming algorithm
  - Country_nationality – conflations between nationalities and countries (“British” maps to “Britain”)
  - Proper Nouns – a list of proper nouns that should not be stemmed.

2.3 Successor Stemmers

- These 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 the distribution of phonemes, the smallest unit of speech that distinguish one word from another.
- 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.

2.3 Successor Stemmers

- These 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 the distribution of phonemes, the smallest unit of speech that distinguish one word from another.
- 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.

Continued.....

- The successor variety of a segment of a word in a set of words is the number of distinct letters that occupy the segment length plus one character.
- Ex: the successor variety for the first three letters (i.e.,word segment)

Continued…..

- These 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 the distribution of phonemes, the smallest unit of speech that distinguish one word from another.
- 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.

Continued…..

- The successor variety of a segment of a word in a set of words is the no of distinct letters that occupy the segment length plus one character.
- Eg: The successor variety for the first three letters (i.e word segment) of a five letter word is the no of words that have the same first three letters but a different fourth letter plus one for the current word .
- The successor varieties of a word are used to segment a word by applying one of the following four methods .

Continued…..

- **Cut off method:** a cut off value is selected to define stem length. The value varies for each possible set of words.
- **Peak and plateau method:** 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.
- **Complete word method:** break on boundaries of complete words.

Continued…..

- **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 of $|D_{ak}|$ is : $H_{ak} = \sum -(|D_{akj}| / |D_{ak}|) (\log_2(|D_{akj}| / |D_{ak}|))$.
  - Using this formula a set of entropy measures can be calculated for a word and its predecessors. A cutoff value is selected and a boundary is identified whenever the cutoff value is reached.

Fig: symbol tree for terms bag, barn, bring, box, bottle, both

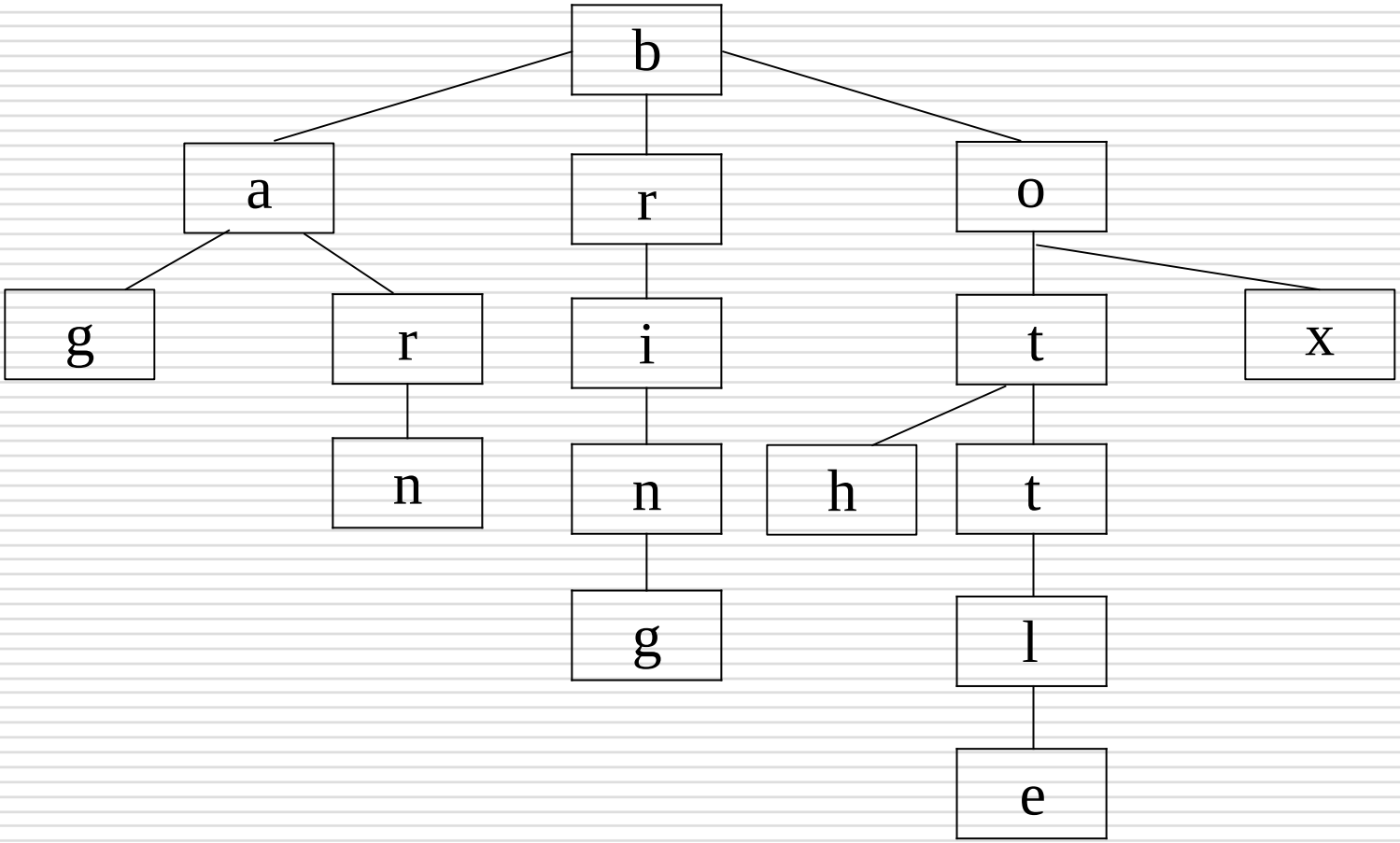


Fig: Successor Variety stemming

using the words in above figure plus the additional word "boxer", the successor variety stemming is shown below

PREFIX	Successor Variety	Branch Letters
b	3	a,r,o
bo	2	t,x
box	1	e
boxe	1	r
boxer	1	blank

Continued…..

- If the cutoff method with value four was selected then the stem would be “boxe”.
- The Peak and Plateau method cannot apply because the successor variety monotonically decreases.
- Applying the complete word method, the stem is “box”.
- The example given does not have enough values to apply the entropy method.
- The advantage of peak and plateau and complete word method is that a cutoff value does not have to be selected.

Continued…..

- After a word has been segmented, the segment to be used as the stem must be selected.
- Hafer and Weiss used the following rule:
  - If (first segment occurs in <=12 words in database)

    first segment is stem

    else (second segment is stem)
- The idea is that if a segment is found in more than 12 words in the text being analyzed, it is probably a prefix.

Contents

- Introduction to data structure
- Stemming Algorithms
- **Inverted File Structure**
- N-Gram data structure
- PAT data structure
- Signature file structures
- Hypertext and XML data structures

3. Inverted File Structure

- The most common data structure used in both DBMS and IRS is Inverted File Structure.
- It minimizes secondary storage access when multiple search terms are applied across the total database.
- All commercial and most academic systems use inversion as the searchable data structure.
- Inverted file structures are composed of three basic files:
  - Document file,
  - Inversion lists (sometimes called posting files)
  - Dictionary

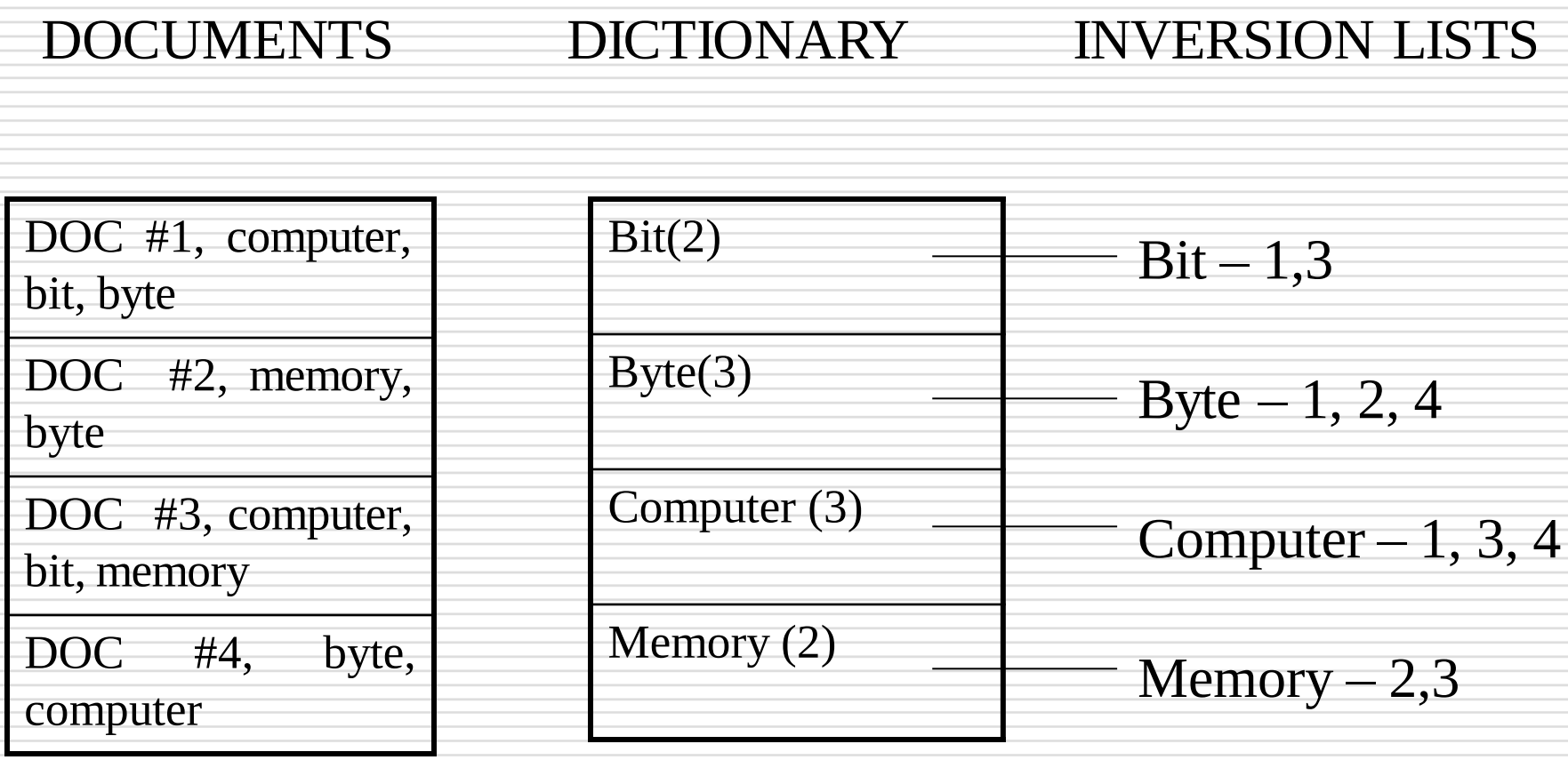
Continued.....

- The name “inverted file” comes from its underlying methodology of storing an inversion of documents:
  - Inversion of document from the perspective that, for each word, a list of documents in which the word is found in is stored (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

Continued…..

- 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 (as shown in below figure).
- Dictionaries can also store other information used in query optimization such as length of inversion lists
- Additional information may be used from the item to increase precision and provide a more optimum inversion list file structure.

Fig 1: Inverted File Structure



Continued.....

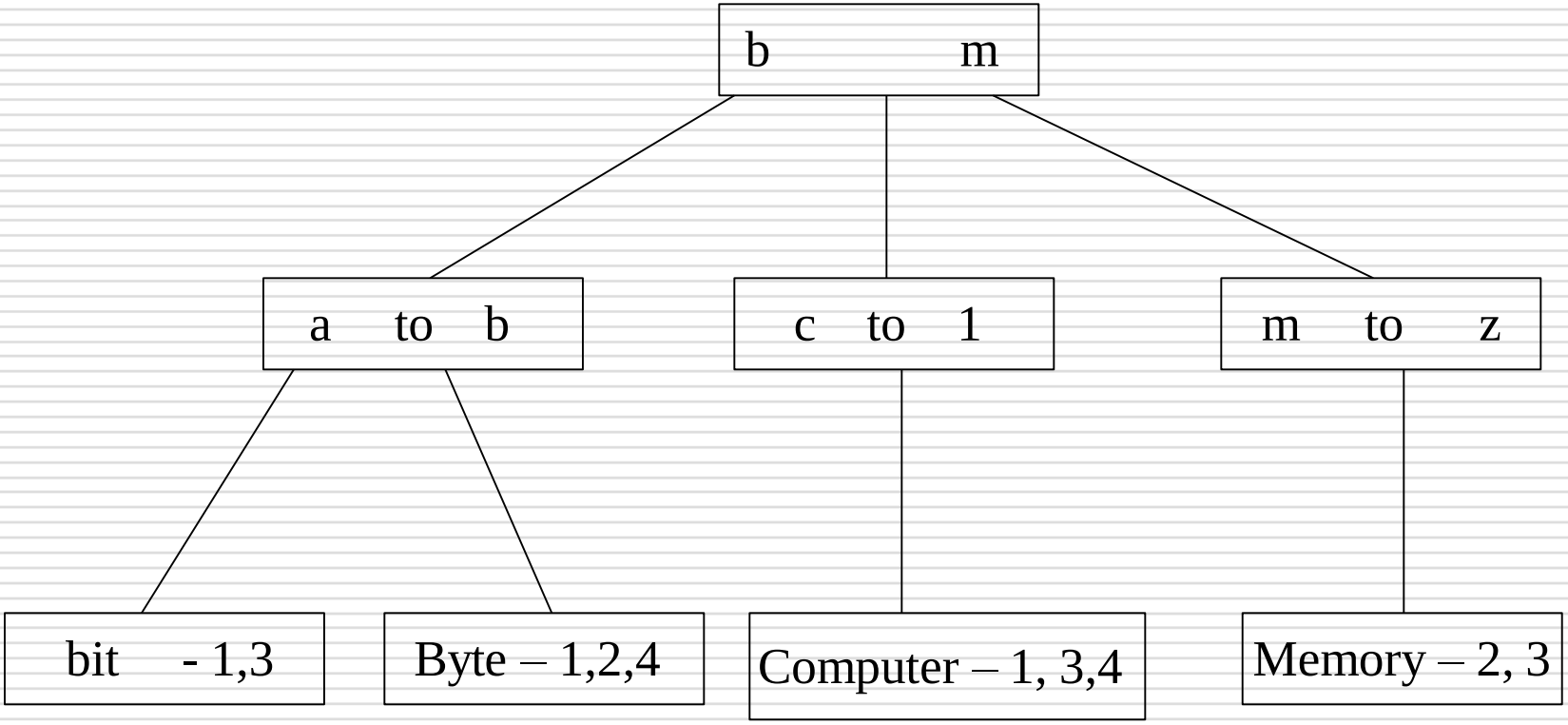
- Ex: if zoning is used, the dictionary may be partitioned by zone. There could be a dictionary and set of inversion lists for the "Abstract" zone in an item and another dictionary and set of inversion lists for the "Main body" zone.
- This increases the overhead when a user wants to search the complete item versus restricting the search to a specific zone.
- If the word "bit" was the tenth, twelfth and eighteenth word in document, then inversion list would appear:

bit – 1(10), 1(12), 1(18)

Continued…..

- Weights can also be stored in inversion lists.
- 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.
- The figure 2 shows how the words in figure 1 would appear.

Fig 2: B-tree inversion lists



Continued.....

- A B-tree of order m is defined as:
  - A root node with between 2 and 2m keys.
  - All other internal nodes have between m and 2m keys
  - All keys are kept in order from smaller to larger.
  - All leaves are kept at the same level or differ by at most one level.
- The nature of information systems is that items are seldom if ever modified once they are produced.
- Most commercial systems take advantage of this fact by allowing document files and their associated inversion lists to grow to a certain maximum size and then to freeze them, starting a new structure

continued.....

- Each of these databases of document file, dictionary, inversion lists is archives and made available for user's query.
- Inversion list file structures are well suited to store concepts and their relationships.
- Inversion lists structures are used because they provide optimum performance in searching large databases
  - The optimality comes from the minimization of data flow in resolving a query.

Contents

- Introduction to data structure
- Stemming Algorithms
- Inverted File Structure
- **N-Gram data structure**
- PAT data structure
- Signature file structures
- Hypertext and XML data structures

4. N-Gram Data Structure

- 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.
- The searchable data structure is transformed into overlapping n-grams, which are used to create search data base.
- Examples of bigrams, trigrams and pentagrams are given in below figure for the word phrase “sea colony”

Fig 1: Bigrams, Trigrams and Pentagrams for "sea colony"

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 #)

Continued….

- For n-grams with n greater than two, some systems allow interword 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 (eg., 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.

Continued....

- Another major use of n-grams is in spelling error detection and correction.
- Damerau specified four categories of spelling errors as shown in following figure.
- Using classification scheme, zamora showed trigram analysis provided a viable data structure for identifying misspellings and transposed characters.
- In information Retrieval, trigrams have been used for text compression and to manipulate the length of index terms.

Fig 2: Categories of Spelling Errors

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

Continued....

- As shown in fig 1, 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 tokens are found.
- The advantage of n-grams is that they place a finite limit on the number of searchable tokens

Continued….

- $MaxSeg_n = (\lambda)^n$

  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)

Continued....

- Because of the processing token bounds of n-gram data structures, optimized performance techniques can be applied in mapping items to an n-gram searchable structure and in query processing.
- There is no semantic meaning in a particular n-gram since it is a fragment of processing token and may not represent a concept.
- Thus n-grams are a poor representation of concepts and their relationships.

Contents

- Introduction to data structure
- Stemming Algorithms
- Inverted File Structure
- N-Gram data structure
- **PAT data structure**
- Signature file structures
- Hypertext and XML data structures

5. 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 original concepts of PAT tree data structures were described as Patricia trees and have gained new momentum as a possible structure for searching text and images.

Continued…

- The name PAT is short for PAtricia Trees (PATRICIA stands for Practical Algorithm To Retrieve Information Coded In Alphanumerics).
- The input stream is transformed into a searchable data structure consisting of substrings.
- 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

Continued…

- 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.
- Some possible sistings for an input text is shown below

Fig: Examples of sistrings

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

Continued…

- 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.

Continued…

- The key values are stored at the leaf nodes (bottom nodes) in the PAT tree.
- For a text input of size “n” there are “n” leaf nodes and “n-1” at most higher level nodes
- Following figure gives an example of the sistrings used in generating a PAT tree.
- If the binary representations of “h” is (100), “o” is (110), “m” is (001) and “e” is (101) then the word “home” produces the input 100110001101…

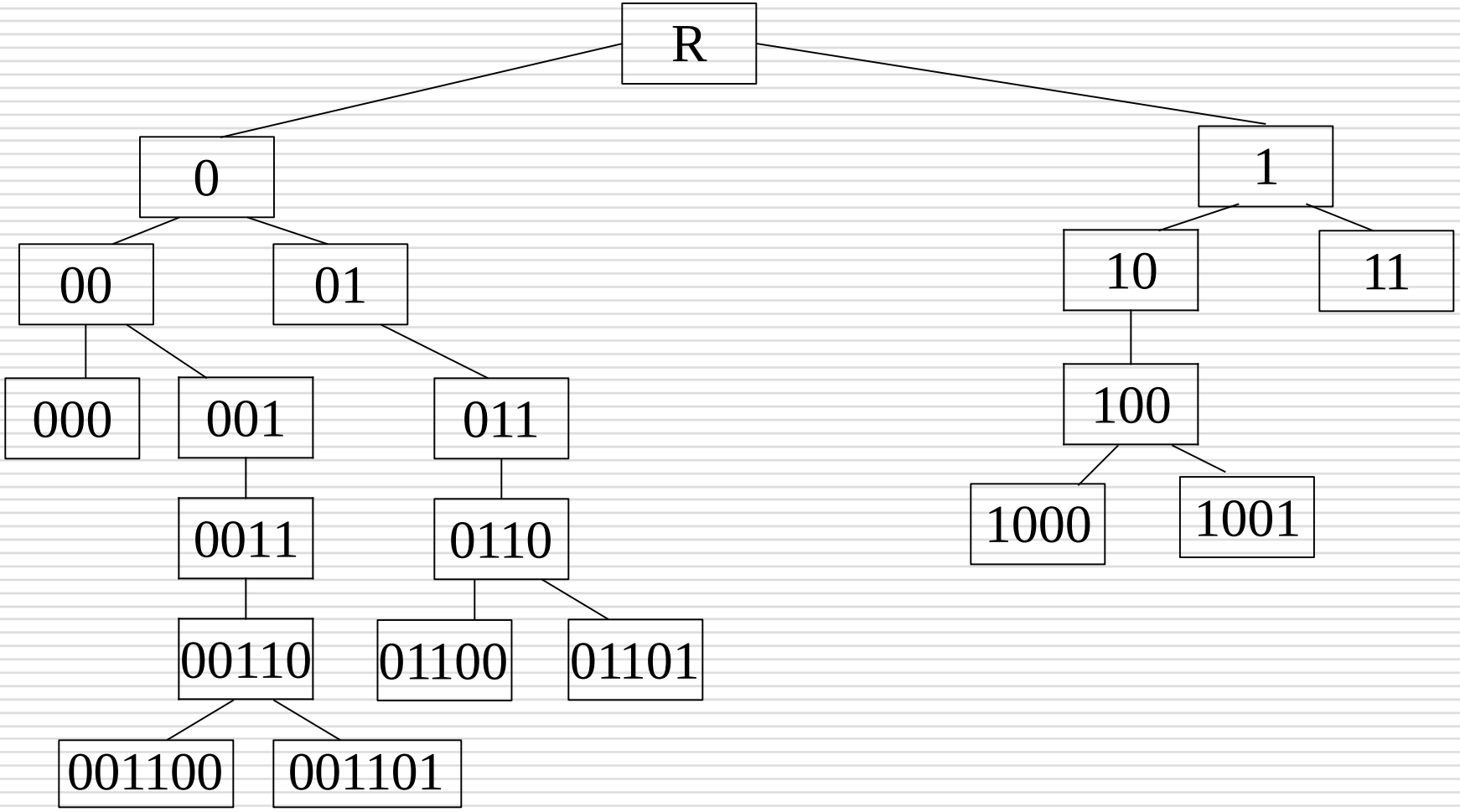
Fig: Sistrings for input "100110001101"

INPUT	100110001101
Sistring 1	1001….
Sistring 2	001100…
Sistring 3	01100…..
Sistring 4	1100…..
Sistring 5	1000…
Sistring 6	000…..
Sistring 7	001101…
Sistring 8	01101….

Continued…

- Using the sistrings, the full PAT binary tree is shown in following figure.

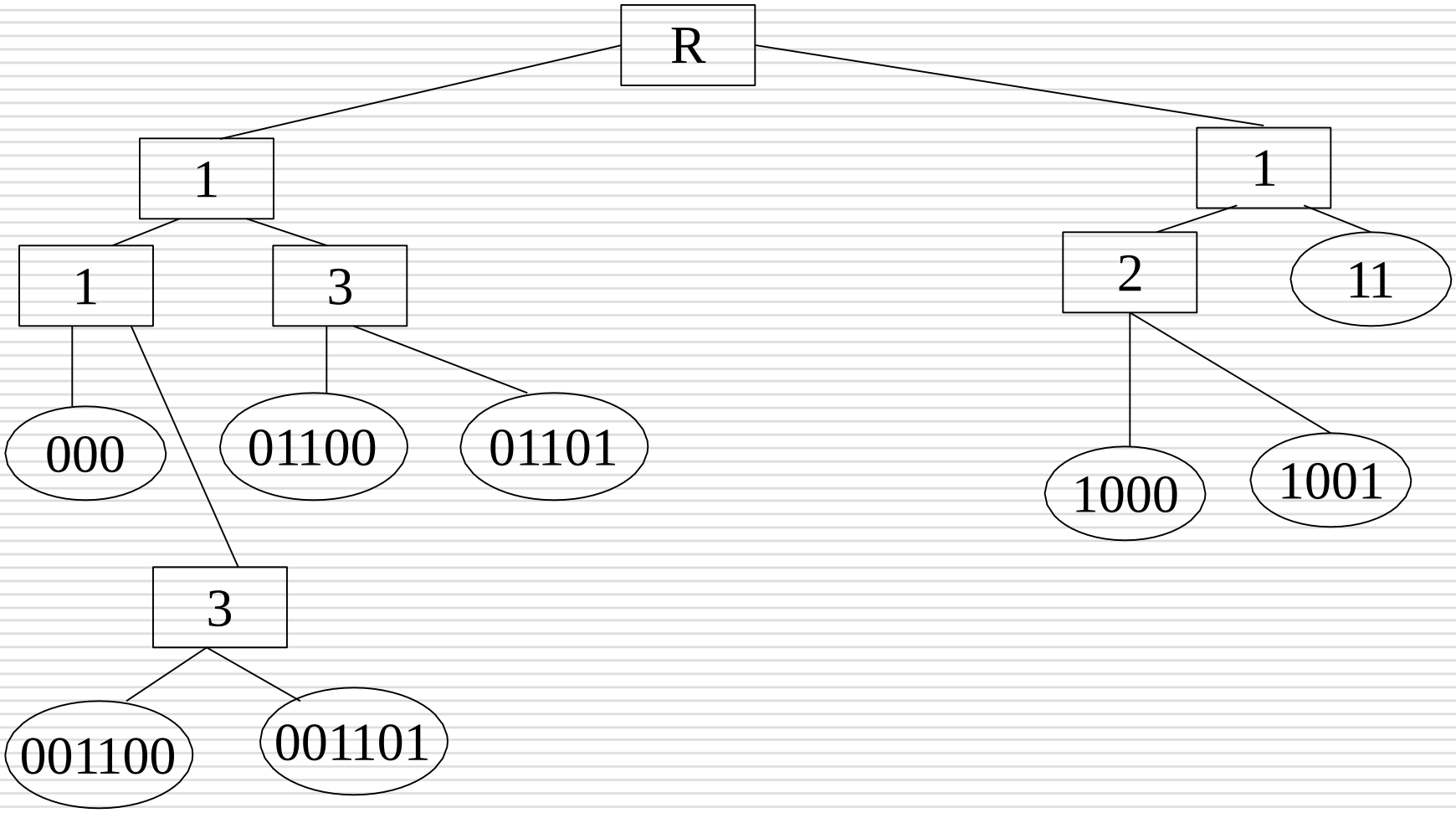
Fig: PAT Binary tree for input "100110001101"



Continued…

- A more compact tree where skip values are in the intermediate nodes is shown in figure below.

Fig: PAT tree skipping bits for "100110001101"



Contents

- Introduction to data structure
- Stemming Algorithms
- Inverted File Structure
- N-Gram data structure
- PAT data structure
- **Signature file structure**
- Hypertext and XML data structures

6. Signature File Structure

- The goal of signature file structure is to provide a fast test to eliminate the majority of items that are not related to query.
- The items that satisfy the test can either be evaluated by another search algorithm to eliminate additional false hits.
- The text of the items is represented in a highly compressed form that facilitates the fast test.
- Because file structure is highly compressed and unordered, it requires significantly less space than an inverted file structure and new items can be concatenated to the end of the structure Vs the significant inversion list update.
- Since items are seldom deleted from information databases, it is typical to leave deleted items in place and mark as deleted.

Continued…

- Signature file search is a linear scan of the compressed version of items producing a response time linear w.r.to file size.
- The surrogate signature search file is created via superimposed coding.
- The coding is based upon words in the item. The words are mapped into a “word signature”.
  - A word signature is a 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 the signature of an item

Continued…

- 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.
- In the following figure 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.

Fig: Superimposed Coding

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

Continued…

- The words in a query are mapped to their signature.
- The signature file can be stored as a signature with each row representing a signature block.
- Associated with each row is a pointer to the original text block.
- A design objective of a signature file system is trading off the size of the data structure Vs the density of the final created signatures.
- Search of the signature matrix requires O(N) search time. To reduce the search time the signature matrix is partitioned horizontally.

Continued…

- Another implementation approach takes advantage of the fact that searches are performed on the columns of signature matrix, ignoring those columns that are not indicated by hashing of any of search terms.
- Thus the signature matrix may be stored in column order Vs row order, called vertical partitioning. This is in effect storing the signature matrix using an inverted file structure.
- Signature files provide practical solution for storing and locating information in a number of different situations.
- Signature files have been applied as medium size databases, databases with low frequency of terms, WORM devices, parallel processing machines and distributed environments (Faloutsos-92)

Contents

- Introduction to data structure
- Stemming Algorithms
- Inverted File Structure
- N-Gram data structure
- PAT data structure
- Signature file structures
- **Hypertext and XML data structures**

Hypertext and XML Data Structures

- The advent of 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 stored in Hypertext Markup Language (HTML) and eXtensible Markup Language (XML).
- HTML is an evolving standard as new requirements for display of items on the Internet are identified and implemented.
- Both of these languages provide detailed descriptions for subsets of text similar to Zoning (in 1st chapter)
- These subsets can be used the same way zoning is used to increase search accuracy and improve display of hit results.

Definition of Hypertext Structure

- The Hypertext data structure is used extensively in the Internet and requires an electronic media storage for the item.
- Hypertext allows one item to reference another item via an imbedded pointer.
- Each separate item is called a node and the reference pointer is called a link.
- The referenced item can be of the same or a different data type than the original (ex: a textual item references a photograph).
- Each node is displayed by a viewer that is defined for the file type associated with the node.

Definition of Hypertext Structure

- Hypertext Markup Language (HTML) defines the internal structure for information exchange across the WWW on the Internet.
- A document is composed of the text of the item along with HTML tags that describe how to display the document.
- Tags are formatting or structural keywords contained between less-than , greater than symbols (eg: <title>,)
- The HTML tag associated with hypertext linkages is

  <a href= ….#NAME /a> where “a” and “/a” are an anchor start tag and anchor end tag denoting the text that the user can activate.

Definition of Hypertext Structure

"href" is the hypertext reference containing either a file name if the referenced item is on this node or an address (URL) and a file name if it is on the other node.

"#NAME" defines a destination point other than the top of th e item to go to.

XML

- The eXtensible Markup Language is starting to become a standard data structure on the WEB.
- Its objective is extending HTML with semantic information
- The W3C(World Wide Web Consortium) is redeveloping HTML as a suite of XML tags.
- Hypertext links for XML are being defined in the Xlink (XML Linking Language) and Xpoint (XML Pointer Language) specifications.