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.

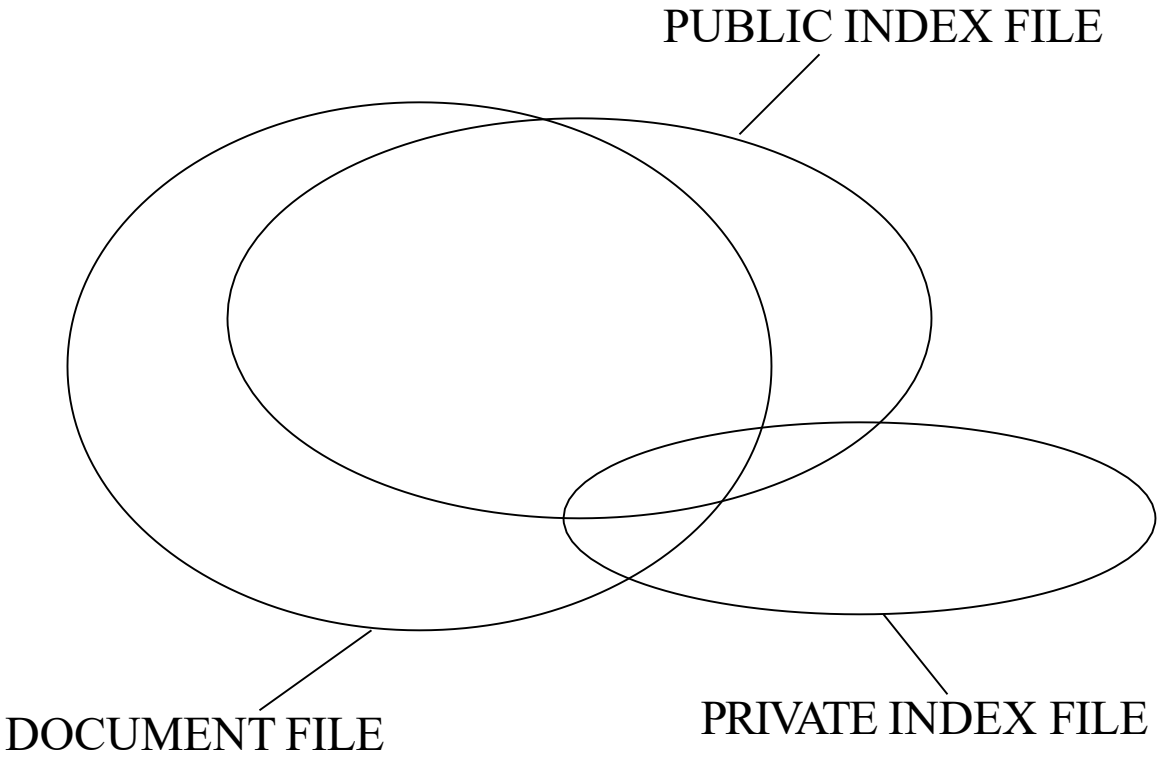
Contents

- 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.
- 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.
- **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…

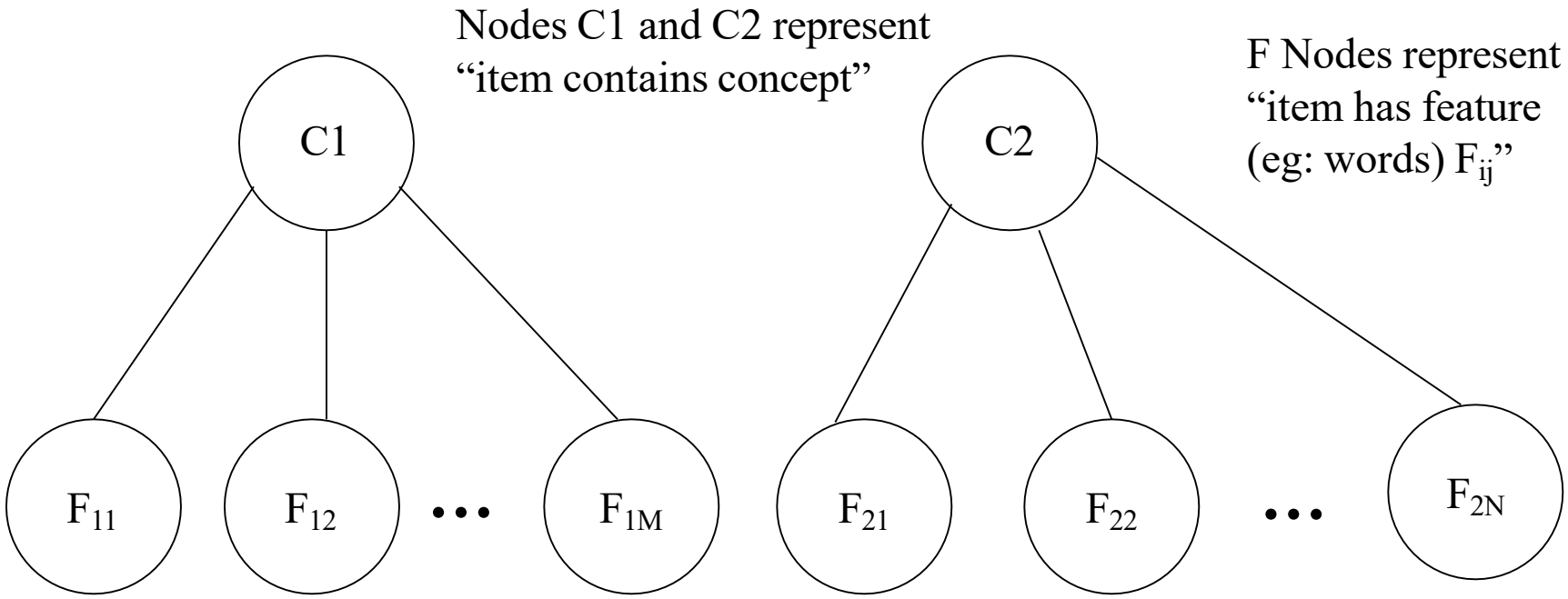


Fig: Two level Bayesian Network

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:

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

- 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:

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

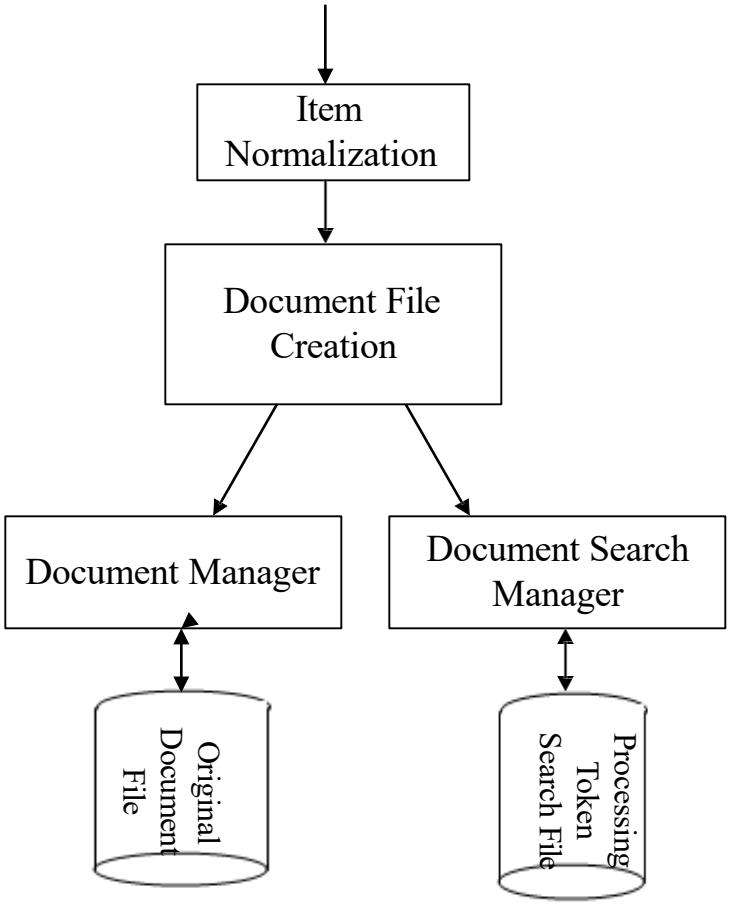
Contents

- 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

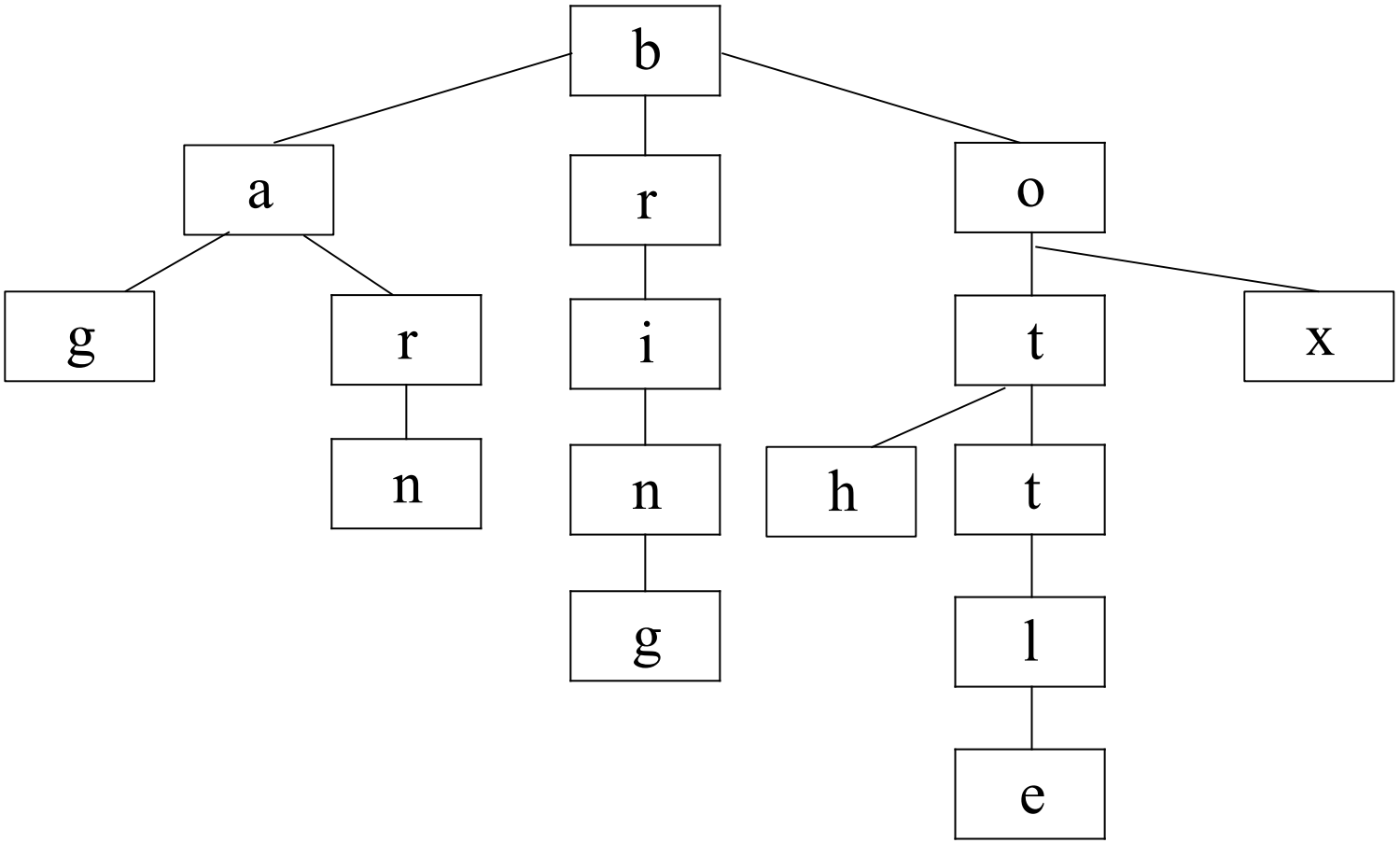


Fig: Successor Variety stemming

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

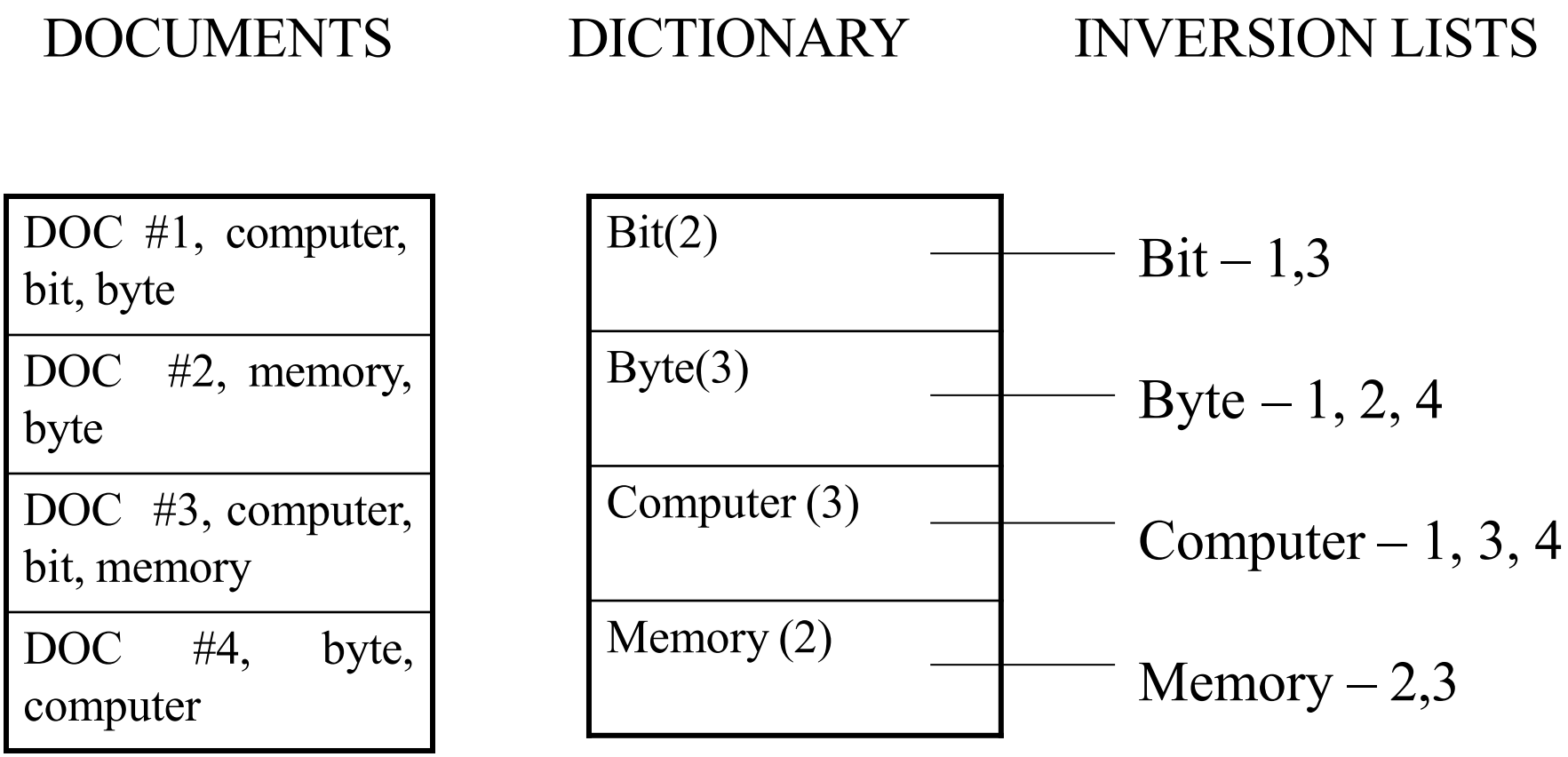
Continued.....

- 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…

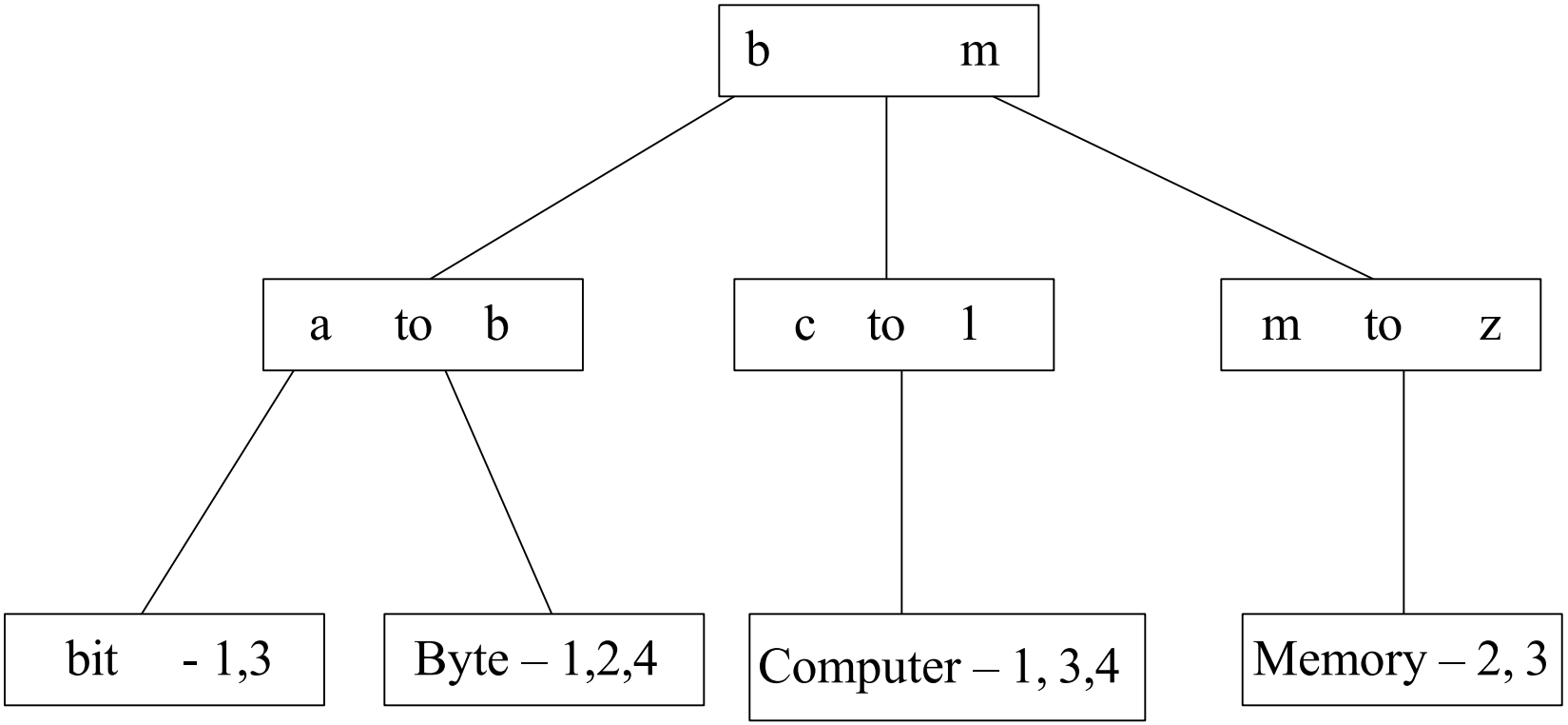
- Ex: if zoning is used, the dictionary may be partitioned by zone. There could be a dictionary and set of inversion lists for the "Abstract" zone in an item and another dictionary and set of inversion lists for the "Main body" zone.
- This increases the overhead when a user wants to search the complete 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:

$$\text{bit} - 1(10), 1(12), 1(18)$$

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…..

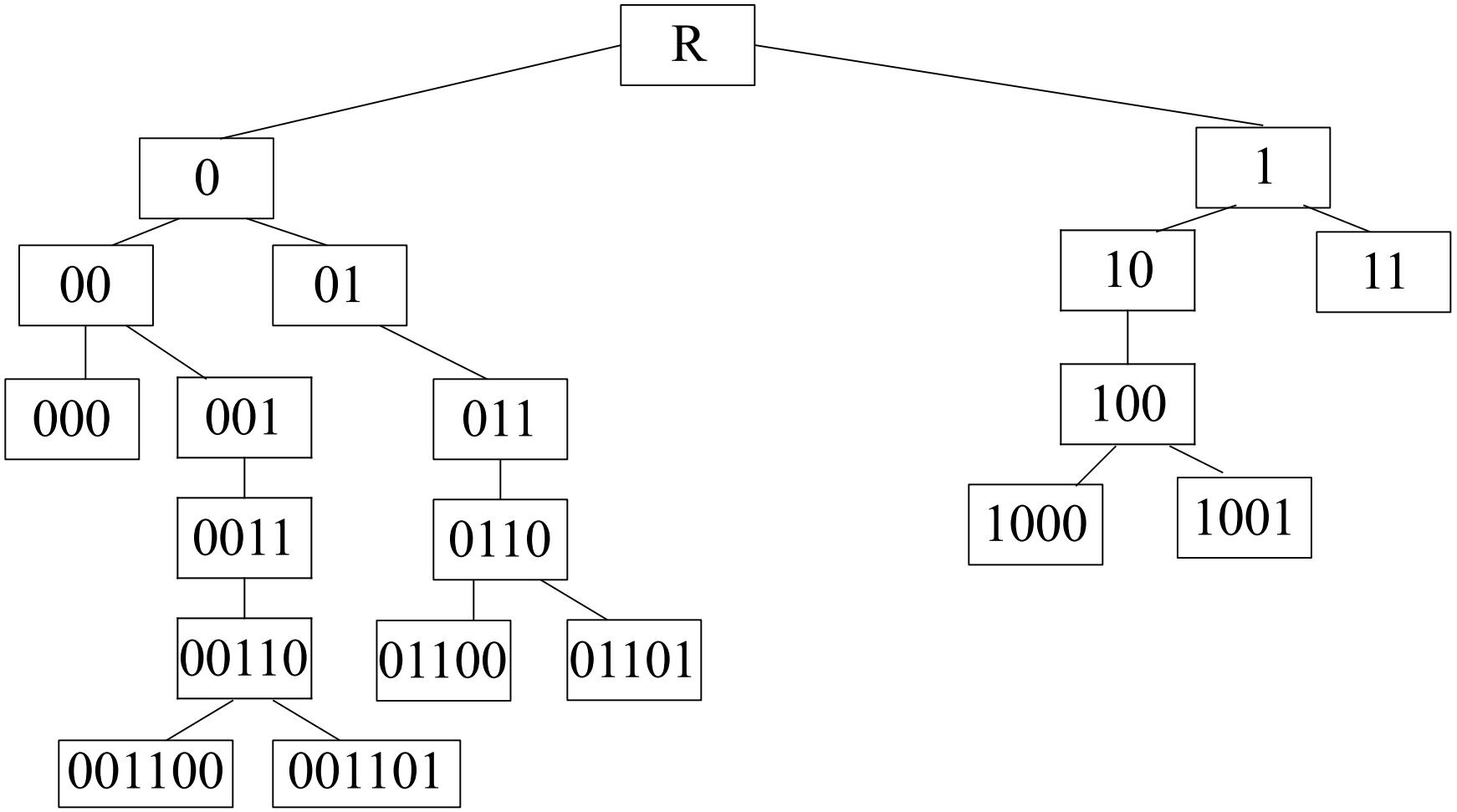
Sistring 7 001101…

Sistring 8 01101….

Continued…

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

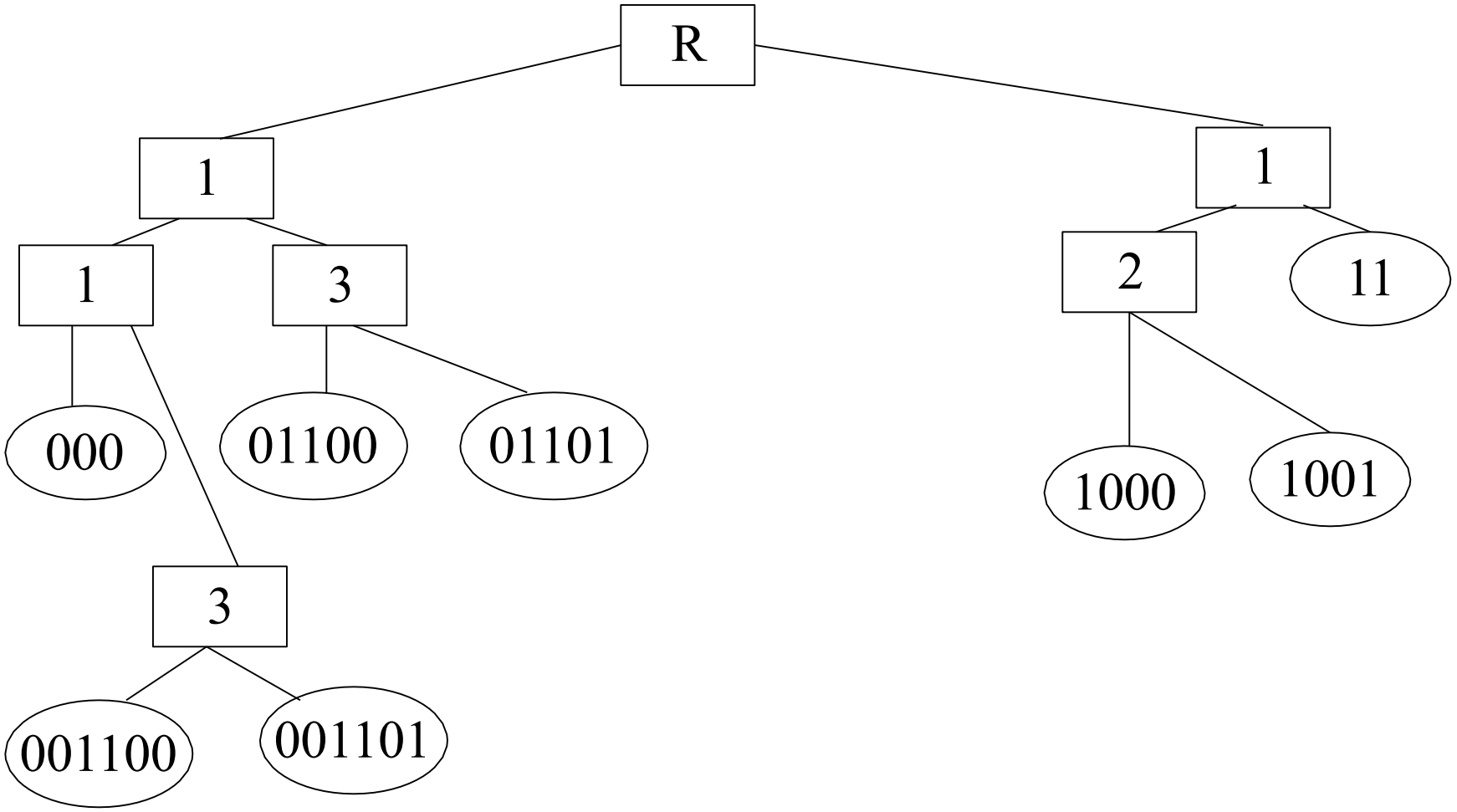
Fig: PAT Binary tree for input "100110001101"



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.