UNIT - 3

Automatic Indexing

Automatic Indexing:

1. Classes of Automatic Indexing
2. Statistical Indexing
3. Natural Language
4. Concept Indexing
5. Hypertext linkages

Document and Term clustering:

6. Introduction to clustering
7. Thesaurus Generation
8. Item clustering
9. Hierarchy of clusters

Automatic Indexing:

It focuses on the process and algorithms to perform indexing. The indexing process is a transformation of an item that extracts the semantics of the topics discussed in the item. The extracted information is used to create the processing tokens and the searchable data structure. The semantic of the item not only refers to the subjects discussed in the item but also in weighted systems, the depth to which the subject is discussed. The index can be based on the full text of the item, automatic or manual generation of a subset of terms to represent the item, natural language representation of the item or abstraction to concepts in the item.

1. classes of Automatic Indexing:

Automatic Indexing is the process of analyzing an item to extract the information to be permanently kept in an index. This process is associated with the generation of the searchable data structures associated with an item.

The below fig shows is expanded to show where the search process relates to the indexing process.

The left side of the fig including Identify Processing tokens, Apply stop lists, Characterize tokens, Apply stemming & Create searchable data structure is all part of the indexing process.

All systems go through an initial stage of zoning & identifying the processing tokens used to create the index. Some systems automatically divide the document up into fixed length passages or localities, the indexed filters, such as stop lists & stemming algorithms, are

frequently applied to reduce the number of tokens to be processed. The next step depends upon the search strategy of a particular system.

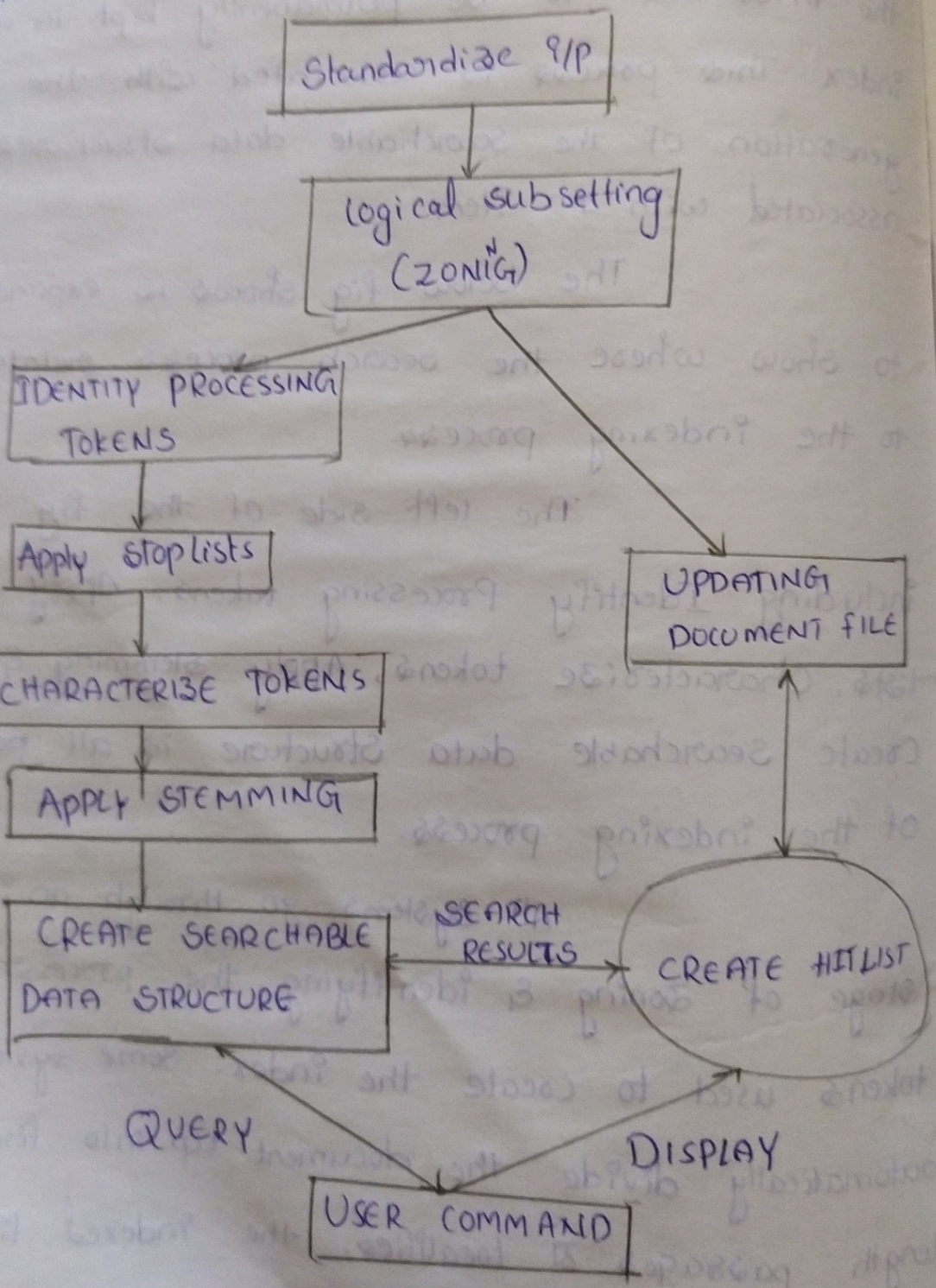


fig: Data Flow in Information Processing System

Search str
natural lan
structure

Statistical

broadest r
most prev
words/ phr
values.
natural

lar proce
techniques
levels of
eg: pre
Concept

within
in the
specific

defined
linkages
items
an ite

Search strategies can be classified as statistical, natural language, & concept. An index is the data structure created to support the search strategy

Statistical Indexing & strategies:

Its cover the broadest range of indexing techniques & are the most prevalent in commercial systems. The words/ phrases are the domain of searchable values.

Natural language:

It approaches perform the simi lar processing token identification as in statistical techniques, but then additionally perform varying levels of natural language parsing of the item eg: present, past, future actions.

Concept index:

Concept indexing uses the words within an item to correlate to concepts discussed in the item. This is a generalization of the specific words to values used to index the item.

Finally a special class of indexing can be defined by creation of hypertext linkages. These linkages provide virtual threads of concepts b/w items Vs directly defining the concept within an item.

2. Statistical Indexing:

It uses frequency of occurrence of events to calculate the num to indicate potential relevance of an item. The documents are found by normal Boolean search and then statistical calculation are performed on the hit file, ranking the o/p (eg: term-frequency algorithm)

1. probability weighting
2. Vector weighting
   - > Simple Term Frequency algorithm
   - > Inverse Document Frequency algorithm
   - > Signal weighting
   - > Discrimination value
   - > problems with the weighting schemes
   - > problems with the vector model
3. Bayesian model.

1. Probabilistic weighting:

Probabilistic systems attempt to calculate a probability value that should be invariant to both calculation method & text corpora (Large collection of written/spoken texts).

The probabilistic approach is based on direct application of theory of probability to IRS.

Advantage - Uses the probability theory to develop the algorithm. This allows easy integration of the final results when searches are performed across multiple databases & use different search algorithms.

The use of probability theory is a natural choice - the basis of evidential reasoning (drawing conclusions from evidence).

This is summarized by PRP (Probability Ranking Principle) & its plausible corollary (reasonable result).

PRP - hypothesis - If a reference retrieval systems response to each request is a ranking of the documents in order of decreasing probability of usefulness to the user, the overall effectiveness of the system to the users is best obtainable on the basis of the data available.

Plausible corollary: The techniques for estimating the probabilities of usefulness for o/p ranking

in IR is standard probability theory & Statistics probabilities are based on binary condition the item is relevant or not. But in information sys the relevance of an item is a continuous fun from non-relevant to absolutely useful.

Source of problems:

In application of probability theory come from lack of accurate data & Simplified assumptions that are applied to mathematical modeling.

Cause the results of probabilistic approaches in ranking items to be less accurate than other approaches.

Advantage of probabilistic approach is that it can identify its weak assumptions & work to strengthen them.

There are many different areas in which the probabilistic approach may be applied. Ex: logistical regression.

The approach starts by defining a "model 0" system.

In retrieval system there exist query

q; and a
of attributes
of term freq
(ex: counts
and from t
s in the da
ts indexed

a random
for which
made from

odds (logod
is present

log(O(R

relevant to

logodds

log(O(R|

applied to
a docum

q_i and a document term d_j which has a set of attributes $(v_1 \dots v_n)$ from the query (ex: counts of term frequency in the query), from the document (ex: counts of term frequency in the document) and from the database (ex: total num of documents in the database divided by the num of documents indexed by the term).

The logistic reference model uses a random sample of query document terms for which binary relevance judgment has been made from judgment from samples.

Logarithm O is the logarithm of the odds (logodds) of relevance for terms t_k which is present in document D_j and query Q_i

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

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

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

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

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

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

Additional attributes of relative frequency in the query (QRF), relative frequency in the document (DRF) and relative frequency of the term in all the documents (RFAD) were included, producing the logodds formula:

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

QRF = QAF \ (total num of terms in the query)

DRF = DAF \ (total num of words in the document)

RFAD = (total num of term occurrences in the database) \ (total num of all words in the database).

Logs are used to reduce the impact of frequency information; then smooth out skewed distributions.

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

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

The logistic inference method was applied to the test database along with the cornell SMART vector system, inverse document frequency & cosine relevance weighting formulas.

The logistic inference method outperformed the vector method. Attempts have been made to combine different probabilistic techniques to get a more accurate value.

This combination of logarithmic odds has not presented better results. The objective is to have the strong points of different techniques compensate for weaknesses.

To date this combination of probabilities using averages of logodds has not produced better results and in many cases produced worse results.

2. Vector Weighting:

Earliest system that investigated statistical approach is SMART system of Cornell university. The system is based upon a vector model. A vector is one-dimensional set of values, where the position of each value in the set is fixed & represents a particular domain.

Each position in the vector represents a processing token. There are 2 approaches to the domain of values in the vector: binary or weighted under binary approach the domain contains a value of 1 or 0. [∵ 1 represented existence of processing tokens in the item].

In the weighted approach, the domain is set of +ve value. The value for each processing token represents the relative importance of the item.

	Petroleum	mexico	oil	Taxes	Refineries	Shipping
Binary	(1 ,	1 ,	1,	0 ,	1 ,	0)
weighted	(2.8 ,	1.6 ,	3.5,	3 ,	3.1 ,	.1)

fig: Binary & Vector Representation of an Item.

The above fig shows how an item that discusses petroleum refineries in mexico would be represented.

A weighted vector acts the same as a binary vector but it provides a range of values that accommodates a variance in the value of relative importance of processing tokens in representing the item.

The use of weights also provides a basis for determining the rank of an item.

The vector approach allows for a mathematical & a physical representation using a vector space model. Each processing token can be considered another dimension in an item representation space. In below fig shows a 3-dimensional vector representation assuming there were only 3 processing tokens, Petroleum, Mexico & oil.

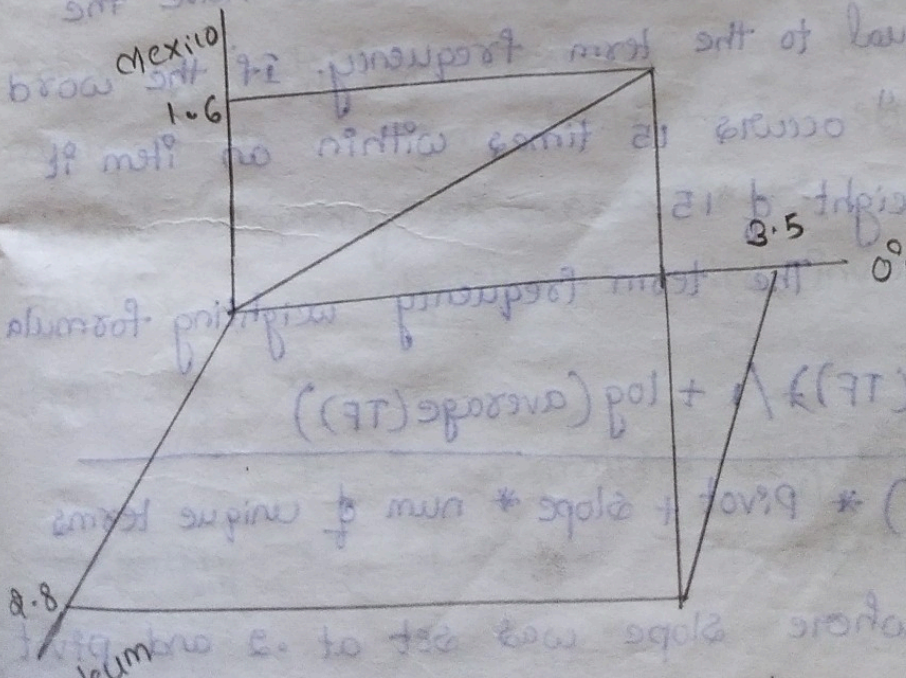


fig: Vector Representation

i) Simple term frequency algorithms:

In both the unweighted & weighted approaches, an automatic indexing process implements an algorithm to determine the weight to be assigned to a processing token for a particular item.

In statistical systems, the data that are potentially available for calculating a weight are the frequency of occurrence of the processing token in an existing item (i.e., term frequency - TF), the frequency of occurrence of the processing token in the existing database (i.e., total frequency - TOTF) & the number of unique items in the database that contain the processing token.

Simplest approach is to have the weight equal to the term frequency. If the word "computer" occurs 15 times within an item it has a weight of 15.

The term frequency weighting formula:

$$\frac{(1 + \log(TF)) / 1 + \log(\text{average}(TF))}{(1 - \text{slope}) * \text{Pivot} + \text{slope} * \text{num of unique terms}}$$

where slope was set at .2 and pivot was set to the average no of unique terms

occuring in the collection. Slope & pivot are constants for any document/query set. This leads to the final algorithm that weights each term by the above formula divided by the pivoted normalization:

((1+log(TF))/(1+log(average(TF))/(slope)(No. unique terms) + (1-slope) * (pivot)

(ii) Inverse Document frequency (IDF):

One of the objectives of indexing an item is to discriminate the semantics of that item from other items in the database. If the token computer represents a concept used in an item, but it does not help a user find the specific information being sought since it returns the complete DB.

This leads to the general stmt enhancing weighting algorithms that the weight assigned to an item should be inversely proportional to the frequency of occurrence of an item in the database. This algorithm is called inverse document frequency (IDF).

The un-normalized weighting formula is:

$$WEIGHT_{ij} = TF_{ij} * [\log_2(n) - \log_2(IF_{ij}) + 1]$$

where $WEIGHT_{ij}$ is the vector weight that is assigned to term "j" in item "i",

TF_{ij} (term frequency) is the frequency of term j in item i, n is the number of items in the database, and IF_{ij} (item frequency or document frequency) is the num of items in the database

Ex: Assume that the term "oil" is found in 128 items, "mexico" is found in 16 items and "refinery" is found in 1024 items.

If a new item arrives with all 3 terms in it, "oil" found 4 times, "mexico" found 8 times, and "refinery" found 10 times & there are 2048 items in the total database.

weight calculations using inverse document frequency.

$weight_{oil} = 4 * (\log_2(2048) - \log_2(128) + 1) = 4 * (11 - 7 + 1) = 20$

$weight_{mexico} = 8 * (\log_2(2048) - \log_2(16) + 1) = 8 * (11 - 4 + 1) = 64$

$weight_{refinery} = 10 * (\log_2(2048) - \log_2(1024) + 1) = 10 * (11 - 10 + 1) = 20.$

with the resultant inverse document frequency item vector = (20, 64, 20).

The value of "n" and IF vay as items are added and deleted from the database.

(iii) Signal

adjusts th

based upo

term in

frequency

the item

ability

for exa

"DRILL"

frequencie

Item

In

value

the pro

the tim

instance

(iii) Signal weighting:-

Inverse document frequency adjusts the weight of processing token for an item based upon the num of items that contain the term in the existing database.

It does not account for is the term frequency distribution of the processing token in the item that contain the term - can affect the ability to rank items.

for example: Assume the terms "SAW" and "DRILL" are found in 5 items with the following frequencies:

Item	Distribution	SAW	DRILL
	A	10	2
	B	10	8
	C	10	18
	D	10	10
	E	10	18

In information theory, the information content value of an object is inversely proportional to the probability of occurrence of the item.

An instance of an event that occurs all the time has less information value than an instance of a seldom occurring event.

This is represented as INFORMATION = $-\log_2(P)$, where P is the probability of occurrence of event "P."

The information value for an event that occurs .5% of the time is: INFORMATION = $-\text{Log}_2(.0005)$

$= -(-10)$

$= 10.$

50% of the time is: INFORMATION = $-\log_2(.50)$

$= -(-1)$

$= 1$

If there are many independent occurring events then the calculation for the average information value across the events is:

$$\text{AVE\_INFO} = -\sum_{k=1}^{n} P_k \log_2(P_k)$$

Its value decreases proportionally to increases in variances in the values of P_k can be defined as $TF_{ik} / TOTF_k$

Formula for calculating the weighting factor called Signal can be used:

$$\text{Signal}_k = \log_2(\text{TOTF}) - \text{AVE\_INFO}$$

It producing a final formula of:

$$\text{weight}_{ik} = TF_{ik} * \text{Signal}_k$$

$$Weight_{ik} = TF_{ik} * [\log_2(TOTF_k) - \sum_{i=1}^{n} TF_{ik} / TOTF_k \log_2(TF_{ik}/TOTF_k)]$$

Based an example of item distribution for SAW and DRILL is

$$Signal_{SAW} = \log_2(50) - [5 * \{10/50 \log_2(10/50)\}] = 3.35$$

$$Signal_{DRILL} = LOG_2(50) - [2/50 LOG_2(2/50) + 2/50 LOG_2(2/50) + 18/50 LOG_2(18/50) + 10/50 LOG_2(10/50) + 18/50 LOG_2(18/50)]$$

$$= 3.75$$

The weighting factor for term "DRILL" that does not have a uniform distribution is larger than that for term "SAW" and gives it a higher weight.

This technique could be used by itself or in combination with inverse document frequency of other algorithms. The overhead of the additional data needed in an index & the calculations required to get the values have not been demonstrated to produce better results.

It a good example of use of information Theory in developing information retrieval algorithms.

(iv) Discrimination Value:

Another approach to creating a weighting algorithm is based on the discrimination value of the term. The all items appear the same, the harder it is to identify those that are needed.

Salton & Yang proposed a weighting algorithm that takes into consideration the ability for a search term to discriminate among items. They proposed use of a discrimination value for each term "i".

$$DISCRIM_i = AVESIM_i - AVESIM$$

where AVESIM is the average similarity b/w every item in the database & AVESIM$_i$ is the same calculation except that term "i" is removed from all items.

DISCRIMi value being +ve, close to 0/-ve.

» A +ve value indicates that removal of term i has increased the similarity b/w items. In this case, leaving the term in the database assists in discriminating b/w items and is of value.

» A value close to zero, implies that the term's removal or inclusion does not change the similarity.

similarity b/w items.

» If the value is negative, the terms effect on the database is to make the items appear more similar since their average similarity decreased with its removal.

Once the value of DISCRIM is normalized as a positive number, it can be used in the standard weighting formula as:

$$Weight_{ik} = TF_{ik} * DISCRIM_k$$

v) Problems with Weighting Schemes:

Often weighting schemes use information that is based upon processing token distributions across the database. The information database tends to be dynamic with new items always being added & to a lesser degree old items being changed or deleted. Thus these factors are changing dynamically.

There are a num of approaches to compensate for the constant changing values.

» Ignore the variances & calculate weights based upon current values, with the factors changing over time. Periodically rebuild the complete search database.

» Use a fixed value while monitoring changes in the factors. when the changes reach a certain threshold, start using the new value & update all existing vectors with the new value.

» Store the invariant variables (ex: term frequency within an item) & at search time calculate the latest weights for processing token in items needed for search terms.

In the 1st approach, Periodically the database & all term weights are recalculated based upon the most relevant recent updates to the database.

In the 2nd approach, there is a recognition that for the most frequently occurring items, the aggregate values are large. As such, minor changes in the values have negligible effect on the final weight calculation.

The 3rd approach is the most accurate. The weighted values in the database only matter when they are being used to determine items to return from a query.

Solutions: If the system is using an inverted file search structure, this overhead is very minor.

The best
a query ag
different
weighting
the result

vi) Problem

comes in
topics bei
for exam
discussio
Pennsylv
mechanis
its pari
correlatio
in a v
Thus
search

Space
with

changes
h a
value &
new value.
frequency
late the
items
lly the
ulated based
to the
a recognition
items, the
minor changes
on the final
most accurate.
only matter
ine items
inverted
very minor.

The best environment would allow a user to run a query against multiple different time periods & different databases that potentially use different weighting algorithms, & have the system integrate the results into a single ranked Hit file.

vi) Problems with the vector model:

A major problem comes in the vector model when there are multiple topics being discussed in a particular item. For example, assume that an item has an in-depth discussion of "oil" in "mexico" and also "coal" in "Pennsylvania". The vector model does not have a mechanism to associate each energy source with its particular geographic area.

There is no way to associate correlation factors b/w terms since each dimension in a vector is independent of the other dimension. Thus the item results in a high value in a search for "coal in mexico".

Another major limitation of a vector space is in associating positional information with a processing item. term.

The concept of a vector space allows only one scalar value to be associated with each processing term for each item.

3. Bayesian model:

One way of overcoming the restrictions inherent in a vector model is to use a Bayesian approach to maintaining information on processing tokens. The Bayesian model provides a conceptually simple yet complete model for information systems. The Bayesian approach is based upon conditional probabilities.

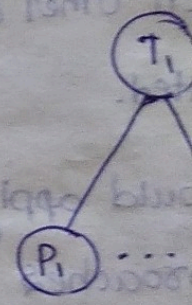
This general concept can be applied to the search fun^n as well as to creating the index to the database. The objective of information systems is to return relevant items.

Thus, the general case, using the Bayesian formula is P(REL/Doc j, Query j) which is interpreted as the probability of relevance (REL) to a search statement given a particular document and query.

In addition to search, Bayesian formulas can be used in determining the weights associated with a particular processing token in an

item. The
item is to
the item.
mine the
& their

of the
ce of to
represents
of proces



"m"
to the
made
Assump
the pr
each of

item. The objective of creating the index to an item is to represent the semantic information in the item. A Bayesian n/w can be used to determine the final set of processing tokens (called topic) & their weights.

In below fig shows a simple view of the process where T_i represents the relevance of topic "i" in a particular item and P_j represents a statistic associated with the event of processing token "j" being present in the item.

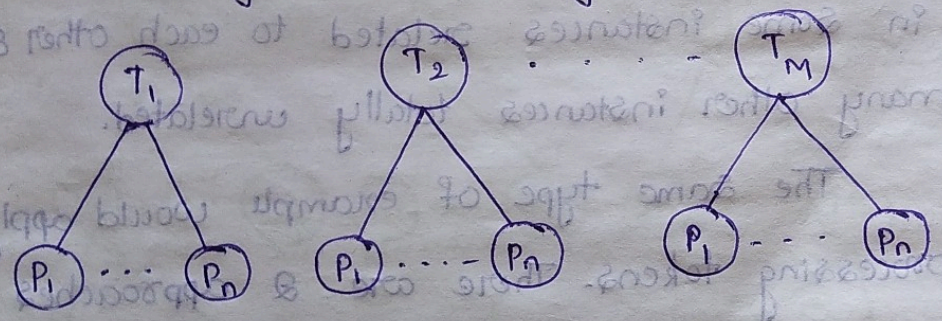


Fig: Bayesian Term Weighting

"m" topics would be stored as the final index to the item. There is one major assumption made in this model:

Assumption of Binary Independence:

» The topics & the processing token statistics are independent of each other.

» The existence of the one topic is not related to the existence of the other topics.

» The existence of one processing token is not related to the existence of other processing tokens.

In most cases this assumption is not true. Some topics are related to other topics & some processing tokens related to other processing tokens.

For example: The topics of "Politics" & "Economics" are in some instances related to each other & in many other instances totally unrelated.

The same type of example would apply to processing tokens. There are 2 approaches to handling this problem.

» The 1st is to assume that there are dependencies, but that the errors introduced by assuming the mutual independence do not noticeably effect the determination of relevance of an item nor its relative rank associated with other retrieved items.

» A 2nd approach can extend the nlw to additional

layers to handle interdependencies.

Thus an additional layer of Independent Topics (ITs) can be placed above the Topic layer & a layer of Independent processing Tokens (IPs) can be placed above the processing token layer.

The below fig shows the extended Bayesian n/w.

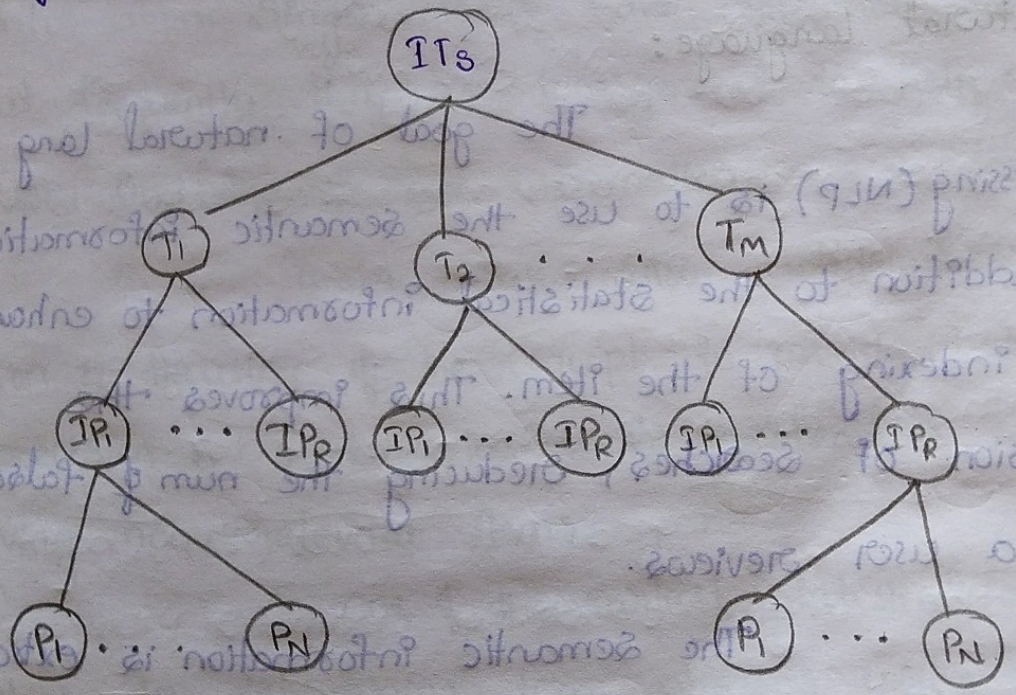


fig: Extended Bayesian n/w.

Extending the n/w creates new processing tokens for those cases where there are dependencies b/w processing tokens. The new set of Independent processing Tokens can be used to define the attributes associated with the set of

topics selected to represent the semantics of an item. To compensate for dependencies b/w topics the final layer of Independent Topics is created. The degree to which each layer is created depends upon the error that could be introduced by allowing for dependencies b/w Topics or Processing Tokens.

3. Natural language:

The goal of natural lang processing (NLP) is to use the semantic information in addition to the statistical information to enhance the indexing of the item. This improves the precision of searches, reducing the num of false hits a user receives.

The semantic information is extracted as a result of processing the lang rather than treating each word as an independent entity. The simplest o/p of this process results in generation of phrases that become indexes to an item. More complex analysis generates thematic representation of events rather than phrases.

Natural Language Processing can also combine the concepts into higher level concepts sometimes referred to as thematic representations.

ii) Index phrase generation:

The goal of indexing is to represent the semantic concepts of an item in the information system to support finding relevant information. Single words have conceptual context, but frequently they are too general to help the user find the desired information.

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

One of the earliest statistical approaches to determining term phrases using of a COHESION factor b/w terms.

$$COHESION_{k,h} = SIZE\text{-}FACTOR * (PAIR\text{-}FREQ_{k,h} / TOTF_k * TOTF_H)$$

where SIZE-FACTOR is a normalization factor based upon the size of the vocabulary & $PAIR\text{-}FREQ_{k,h}$ is the total frequency of co-occur-

-rence of the pair $term_k$, $Term_h$ in the item collection. Co-occurrence may be defined in terms of adjacency, word proximity, sentence proximity etc.

This initial algorithm has been modified in the SMART system to be based on the following guidelines.

» Any pair of adjacent non-stop words is a potential phrase.

» Any pair must exist in 25 or more items.

» phrase weighting uses a modified version of the SMART system single term algorithm.

» Normalization is achieved by dividing by the length of the single term sub-vector.

The tagged text Parser (TTP), based upon the linguistic string Grammar, produces a regularized parse tree representation of each sentence reflecting the predicate - argument structure.

The tagged text parser contains over 400 grammar production rules. Some examples of

CLASS	Examples
determiners	a, the
Singular nouns	paper, notation, language

CLASS	Examples
plural nouns	operations, data, processes
preposition	in, by, of, for
adjective	high, concurrent
present tense verb	presents, associates
present participal	multiprogramming.

To determine if a header-modifier pair warrants indexing, Stozalkowski calculate a value for Informational Contribution (IC) for each element in the pair. The basis behind the IC formula is a conditional probability b/w the terms. The formula for IC b/w 2 terms (x,y) is:

$$IC(x,[x,y]) = f_{x,y} / N_x + D_x - 1$$

where $f_{x,y}$ is the frequency of (x,y) in the database, N_x is the num of pairs in which x occurs at the same position as in (x,y) & $D(x)$ is the dispersion parameter which is the num of distinct words with which x is paired.

When IC = 1, x occurs only with the y ($f_{x,y} = n_x$ and $d_x = 1$).

following formula for weighting phrases:

$$weight(phrase_i) = (C_1 * \log(termf) + C_2 * \alpha(N,i)) * IDF$$

where $\alpha(N,i)$ is 1 for $i<N$ and 0 otherwise

C_1, C_2 normalizing factors.

IDF is inverse document frequency.

(ii) Natural Language Processing (NLP):

NLP not only produces more accurate term phrases, but can provide higher level semantic information identifying relationships b/w concepts. System adds the functional processes Relationship concept Detectors, conceptual Graph Generators & Conceptual Graph matchers that generate higher level linguistic relationships including semantic & is course level relationships.

During the 1st phase of this approach, the processing tokens in the document are mapped to subject codes. These codes equate to index term assignment & have some similarities to the concept-based systems.

The next phase is called the Text structure, which attempts to identify

general discov

The next level

b/w the conc

weights to the

weights are

information &

used in estab

4. Concept Ind

the terms

information

level concepts

a level fur

of terms

a reduced

number of

information

classes cr

general discourse level areas within an item.

The next level of NLP identifies interrelationships b/w the concepts.

The final step is to assign final weights to the established relationships. The weights are based upon a combination of statistical information & values assigned to the actual words used in establishing the linkages.

4. Concept Indexing:

NLP starts with a basis of the terms within an item and extends the information kept on an item to phrases & higher level concepts such as the relationships b/w concepts. Concept indexing takes the abstraction a level further. Its goal is use concepts instead of terms as the basis for the index, producing a reduced dimension vector space.

Concept indexing can start with a number of unlabeled concept classes & let the information in the items define the concepts classes created.

A term such as "automobile" could be associated with concepts such as vehicle, transportation, "mechanical device", "fuel" and "environment". The term "automobile" is strongly related to "vehicle", lesser to "transportation" and much lesser the other terms.

Thus a term in an item needs to be represented by many concept codes with different weights for a particular item. The basis behind the generation of the concept approach is a neural n/w model.

Special rules must be applied to create a new concept class. Example demonstrates how the process could work for the term "automobile."

TERM: automobile

	weights for associated concepts:
vechicle	.65
Transportation	.60
Environment	.35
fuel	.33
mechanical device	.15

Vector representation Automobile: {.65,..,.60...,.35,.33...,.15)

5. Hypertext Link

representation
be generated
retrieval dim
viewed as a

Text

references as

al electronic
the items
linkages all
the linked
issue: ho
locate rel

mechanism
1. manuall
were inf
indexed
user na
the hyp

5. Hypertext Linkages:

It's a new class of information representation is evolving on the Internet. Need to be generated manually creating an additional inform" retrieval dimension. Traditionally the document was viewed as 2 dimensional.

Text of the item as one dimension and references as 2nd dimension.

Hypertext with its linkages to additional electronic items, can be viewed as n/w b/w the items that extend contents, ie by embedding linkages allows the user to go immediately to the linked item.

Issue: How to use this additional dimension to locate relevant information.

At the internet we have 3 classes of mechanism to help find information.

1. Manually generated Indexes - Ex: www.yahoo.com were information sources on the home page are indexed manually into hyperlink hierarchy. The user navigates through hierarchy by expanding the hyper link. At some point the user starts

to see the end of items.

2. Automatically generated indexes – sites like lycos.com and altavista.com automatically go to other internet sites and retwan the text, google.com

3. Web Crawlers: A web crawler (also known as a web spider ∂ web robot) is a program ∂ automated script which browses the world wide web in a methodical, automated manner.

Web crawlers (eg: open text, path finder) are tools that allow a user to define items ∉ interest & they automatically go to various sites on the Internet searching for the desired information.

What is needed is an index algorithm for items that look at hypertext linkages as an extension ∉ the concept where the link exits current concept is defined by the proximity ∉ inform

Approaches in viewing the hyperlinks:

Approach 1:

In this approach the hyperlink is viewed as an extension of the text of item in another dimension.

The index values ∉ hyperlinked item has

reduced

biased

The w

weight i,

where

weight i,

Link i,k

α, β, ∂ →

be eith

level to

index

the h

Approac

genera

→ Sev

this

reduced weighted value from contiguous text biased by the type of linkage.

The weight of processing token could be.

$$Weight_{i,j,k,l} = (\alpha * weight_{i,j} + \beta * weight_{k,l}) * (\gamma * link_{i,k})$$

where

$weight_{i,j,k,l}$ → Weight associated with token j in item i & token l in item k. (all these are related via hyperlink).

$Link_{i,k}$ → Weight related with strength of the link.

α, β, γ → weighting / normalization factors.

The link can be one-level link and it can be either strong or weak or it would be a multi level transitive link.

The values can be stored in an expanded index structure or calculated dynamically if only the hyperlink relationships b/w items are available.

Approach 2:

→ In this approach, the system could generate the hyperlinks b/w the items automatically.

→ Several attempts have been made to achieve this capability but they face problems with

Static Vs dynamic growing databases & the efficiency needed for an operational environment is ignored.

→ A new sol^{n} on the basis of documents segmentation & clustering have been proposed by kelog & subhas

→ The links b/w the document pairs for each cluster are generated with the linkage at both the document & document sub-part level using cover-coefficient based incremental clustering method (C^{3}M)

→ The clustering & automatic link generation is performed in parallel.

→ The candidates for hyperlinking are the item pairs in the same cluster & they should have the similarity above a given threshold.

→ The process can be completed in 2 phases.

> The estimation of num of clusters is calculated in the first phase.

> The items are clustered & links are created in the 2nd phase.

→ All of the link information is stored externally instead of storing it within an item or storing persistent link ID with in an item etc.

on demand,

when the mi

the 3 commo

> Misspellings

> parser prob

> problems o

items was

→ The more

→ This techn

referential

similarity

Document

6. Introducti

has been

libraries.

> clusterin

the topic

> The do

of orga

are s

on demand, HTML items are created.
when the missed links are analyzed by the alg, the 3 common problems discovered are,

> Misspellings or representations of multiple words.
> parser problems.
> problems occurred when definition of subparts of items was attempted.
> The more num of errors are from parsing.
> This technique has maximum effectiveness for referential links which naturally have higher similarity measures.

Document and Term Clustering.

6. Introduction to "clustering":

> The concept of clustering has been in most usage as long as there are libraries.
> clustering is used in the process of combining the topics on the same subject.
> The definition of clustering could be "the process of organizing objects into groups whose members are similar in some way or other."

> The main goal of clustering would be in locating the information required.

> As the clustering helps to locate the inform, it needs to be indexed in organization of items in the libraries & it also need to follow the standards of electronic indexes.

> The origin of clustering started with the "generation of thesauri."

> The word Thesaurus coming from latin word which means "treasure". The thesauri is similar to that of a dictionary.

> The thesauri provides the synonyms & antonyms of the words instead of their definitions.

> The main purpose of thesauruś is to help the authors in vocabulary selection.

> The goal of clustering is to combine the similar objects into group.

> There can be linkages b/w different clusters in clustering.

The "process of clustering" follows the following steps: They are

Step 1: > The clustering effort should be defined

with a d

> The defi determinin

> If thesa can be ta

> The deter should b clustering

> with identify can red

Step 2:

> The n determin be clust

> In th should

> In c zones are us title, mo

> Here errone

with a domain.

> The definition of a domain can be considered as determining the scope of the process.

> If thesaurus is considered, the domain or scope can be taken as "medical terms", "scientific terms" etc.

> The determination of the collection of items that should be clustered can be treated as document clustering. This can be a part of complete database.

> with the defining of a domain, it is easy to identify the objects in clustering process and it can reduce the erroneous data.

Step 2: To determine the attributes of the objects.

> The next step after defining domains is to determine the attributes of the object that should be clustered.

> In thesauries, the specific words in the objects should be determined.

> In case of document clustering, the specific zones with in the item are focused. These zones are used to determine the similarity such as title, main body but not references.

> Here the main purpose is to reduce the erroneous relationships.

Step 3: To determine the strength of relationships b/w attributes.

> The next step is to find the relationships b/w the attributes of the object.
> This would help in deciding whether the object should be in this cluster or other cluster.
> In thesauri, this step would help in determining the words which are synonyms & the strength of their relationships.
> In document clustering, this step may define a fun^n known as "similarity fun^n". This fun^n is defined on the basis of co-occurrences of words which would determine the similarity b/w 2 items

Step 4: To determine the total set of objects & their relationships.

> The total set of object determination is the final step.
> In this step, some kind of algorithm is applied to find the clusters & the items to be assigned to the clusters.

Guidelines for cluster characteristics:

There are some guidelines to be considered on the characteristics of the clusters. They are,

1) Cluster sh
> For each
defined
> There is a
to the def
sometimes
> To redu
with the
> For exa
class c
that it
But the
another
2) There
the
> One
The
retri
> If
object
> So
the
3) Ther
obje

1) Cluster should have well defined Semantic definition:

> For each & every cluster, there should be a well defined Semantic definition.

> There is a chance that the name that is assigned to the definition of the cluster leads to confusion sometimes.

> To reduce the confusion, the clusters are named with the numbers in some systems.

> for example, if some items are clustered into a class called "computers" can lead to a confusion that it also contains the items on main memory. But the items on main memory are clustered into another class known as "Hardware".

2) There should be same order of magnitude for the size of class or cluster:

> One of the main purpose of cluster is expansion. The queries can be expanded or the set of items retrieved can also be expanded.

> If suppose 95% of the cluster is filled with the objects, then it can't be used for expansion.

> So order of magnitude should be maintained for the cluster size.

3) There should be no Domination Among the objects of the clusters: One object in the

cluster should not dominate the other object in the cluster.

Example:

> Assume a theasurus class called "computer" contains the objects called "microprocessor," "286 processor", "386 - processor", "i3 processor" & "pentium".
> If the term "microprocessor" is found 94% & other are found 2% each.
> Then, if we use "microprocessor" as a synonym for "286 processor" then it would result in many errors.
> It would be better if we do not replace the place of "microprocessor".

4) The object can be assigned to multiple classes or not:

> The object once created should be only in the same cluster or can it be assigned to other multiple clusters?
> This question would be treated as a trade off based on specificity & partitioning capability of the object semantics.
> If there is some ambiguity of the language, then the objects can be assigned to multiple classes instead of restricting it to single cluster.
> But this would increase the complexity in cluster creation & cluster maintenance.

There are additional important decisions associated with the generation of thesauri that are not part of item clustering.

Important Decisions in Thesauri Generation: They are,

1) Word coordination approach
2) Word relationships
3) Homograph Resolution
4) Vocabulary Constraints.

1) Word Coordination Approach:

> This approach is used to specify the phrases & individual terms that are to be clustered.
> The process of creating the linkages at the time of creating index is known as pre coordination.
> If the coordination occurs at the time of search, it is known as Post Coordination.

2) Word Relationships:

> The thesaurus generation would include a lot of relationships b/w the words that are possible. Three types of relationships are specified by Aitchison & Gilchrist. They are,

a) Equivalence
b) Hierarchical
c) Non hierarchical

a) Equivalance Relationship:

> The relationships that represents synonyms are known as equivalence relationships.

> They are most common in nature.

> In the creation of thesaurus, the words with significant overlapping but little differences are allowed.

Example: > The words "prints" and "photographs" are taken as synonyms even though the "prints" can also include lithography.

> The words that are having same role are also considered as synonyms. They need not have same meaning necessarily.

> Example: The words "genius" and "mason" can be synonyms with respect to the class of "intellectual capability".

b) Hierarchical Relationship:

> A technique where the class is general term & the entries are the examples of the general term. The relationship b/w the general & specific (examples) can be treated as hierarchical relationship.

Example: The relationship b/w "company" & "employee" can be hierarchical.

c) Non Hierarc

> The other can be, part Synonymy an

> The 2 wo can be trea

> If the w same sens

> The other are child are contras

3) Homogra

> If the meanings, t

Example: "field" o

> The elim unique.

> The syst his are o & the

> When t help of

c) Non Hierarchical:

> The other relationships that are non hierarchical can be, Part wholes, collocation, Paradigmatic, Taxonomy, synonymy and Antonymy.

> The 2 words related to each other with proximity (or closeness) can be treated as collocation.

> If the words are related on the basis of same semantics, then they are paradigmatic.

> The other relationships included in semantic n/w are child of, parent of, Part of All these words are contrasted.

3) Homograph Resolution:

> If the word is having multiple different meanings, then it is called as "Homograph".

Example: The term field can mean either "electric field" or "magnetic field".

> The elimination of homographs by providing a unique meaning would be difficult.

> The system would be in such a way that homograp hs are allowed by providing a unique meaning & the user selects the meaning that he desires.

> When the user enters multiple terms, with the help of those terms, the correct meaning of the

homograph can be identified.

4) Vocabulary constraints:

> The guidelines on normalization & specify that is based on vocabulary are included in the vocabulary constraints.

> The constraints based on normalisation can be stems vs complete words.

> The specificity would eliminate the specific terms and it uses more general terms for class identifiers.

7. Thesaurus Generation:

> From hundreds of years, the generation of clusters manually focuses mainly on generating thesaurus.

> But now a days, all the items are available in electronic form. Due to this the automated & statistical clustering techniques are made available.

> The thesauri which is automatically generated would contain the classes that has the usage of words in the corpora ie., database.

> The classes of clusters does not have any name but are just a group of similar terms.

> If the clus
then it requ

> The other
with the ex
computation
classes.

Basic metho

a thesaurus

(i) Hand
(ii) co-occ
(iii) Header

(i) Hand cr

> The th
help in
domain

> The ge
because
meanings

(ii) Co-occ

> There

Co-o

> The t
related

> If the clusters are to be generated optimally, then it requires intensive computation.

> The other techniques of clustering which starts with the existing clusters can reduce the computation but may not generate the optimum classes.

Basic methods of Generating a Thesaurus:

There are 3 basic methods for generating a thesaurus. They are based on,

(i) Hand crafted
(ii) Co-occurrence
(iii) Header modifier.

(i) Hand crafted:

> The thesauri that is manually generated would help in expanding the query if the thesauri is domain specific.

> The general thesaurus would not help much because of one word having many different meanings.

(ii) Co-occurrence:

> There are several techniques based on Co-occurrence.

> The thesaurus is generated based on the related or associated words.

(iii) Head modifier:

> The relationships b/w the terms in the thesaurus are found based on the linguistic relationships.

> The words that are similar in grammatical contexts are considered to be similar.

> The syntactical structures that are discovered by the linguistic parsing are,

(i) Subject verb (ii) verb object

(iii) Adjective noun (iv) Noun noun.

> A mutual value is calculated for each noun by using a log fun^n & the noun would contain a set of verbs, adjectives and nouns.

> Using the mutual information, a final similarity b/w the words is calculated.

Manual clustering:

The manual clustering process would follow the steps that are described in as steps in the process of clustering.

1) The 1st step in the process would be the determination of domain for clustering.

> The determination of domains would help in reducing the ambiguities that can be caused by homographs & others.

2, The starting generation of in the new thesaurus. In from items are used.

> A concor in alphabeti references found.

3, The cor lie in the included ?

> core sho be includ

- unrela
- The wo values

> The oth useful the ot

a) KWO

> The

2, The starting points for the words for the generation of set of words that are to be included in the new thesaurus are taken from the existing thesauri. In the existing thesauri, the concordances from items that covers the domain & dictionaries are used.

> A concordance would be the listing of words in alphabetical order along with their frequency & references of the items in which these words are found.

3, The cost of construction of manual thesaurus would lie in the selection of the words that should be included in it.

> care should be taken that some words should not be included such as,

- unrelated to the domain of thesaurus.
- The words with very high frequency & no inform' values.

> The other tools may be helpful in determining the useful words if a concordance is used.

the other tools include KWOC, KWIC & KWAC.

a) KWOC: Key word Out of Context

> The another term for concordance is KWOC

b) KWIC : Keyword In Context

> The possible term in the context of phrase is displayed by KWIC.

> KWIC is structured in a way that it can easily identify the location of the term that is considered in a sentence.

c) KWAC : key word and context.

> KWAC would display the keywords on the basis of their context.

Example:

KWOC

TERM	FREQ	ITEM Ids
chips	2	doc2, doc4
computer	3	doc1, doc4, doc10
design	1	doc4
memory	34	doc3, doc4, doc8, doc12

KWIC

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

KWAC

chips co
computer co
design co
memory c

fig: examp

> In the KWIC to in

> The KWIC the meani

> In both editor of t fragment meaning

> The KWO resolving

4, After th clustered lines & i

> With human

KWAC

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

fig: examples of KWOC, KWIC and KWAC.

> In the above fig. character "/" is used in KWIC to indicate the end of the phrase.

> The KWIC and KWAC are useful in determining the meaning of homographs.

> In both the KWIC and KWAC displays, the editor of the thesaurus can read the sentence fragment associated with the term and its meaning is determined.

> The KWOC does not provide any data in resolving this ambiguity.

4, After the selection of words, the words should be clustered on the basis of the relationship guidelines & interpretation of the relationship strength.

> With the help of human judgement or human analysis, this step would also be a

part of art of manual creation of thesaurus.

> The thesaurus finally would undergo several quality assurance reviews and the guidelines & then it is finalized.

Automatic Term clustering:

> There are many techniques for the automatic generation of term clusters to create statistical thesauri.

> The basis of all those techniques is that if the 2 terms co-occur (simultaneously) in the same item frequently then the 2 terms are about the same concept.

> The 2 terms would be differed by the completeness by which the terms are related.

> If the correlation is more complete then the time & computational overhead would also be higher.

> The relationship b/w all the combinations of n unique words with an overhead of $O(n^2)$ is computed by the most complete process.

> All the other techniques would be initiated by the arbitrary set of clusters & the process would be repeated on the assignment of terms to these clusters.

> A hierarchi
clusters gen
by making
point.
steps for
of cluste
methods:
atic term
i) complete
ii) cluster
iii) one p
i) Complete
clusterin
of simil
uses ve
> The v
this m
> The vec
a matri
> The ro

> A hierarchy can be created when the num of clusters generated are very large. It can be done by making the initial clusters as the starting point.

steps for processing:

Refer steps in the process of clustering.

Methods:

The different methods used for automatic term clustering are,

(i) Complete Term Relation method.

(ii) clustering using existing clusters.

(iii) One pass Assignments.

(i) Complete Term Relation method:

> In this method of clustering, the clusters are formed on the basis of similarity blw every pair of terms. This uses vector model for understanding.

> The vector model can be used for understanding this method easily.

> The vector model is indicated in the form of a matrix.

> The rows of the matrix represents individual

items & the columns of the matrix represents unique words or processing tokens.

> The values of the matrix would represents how strongly a particular word would represent the concepts in the item.

> The example of a database is shown in below table.

	Term 1	2	3	4	5	6	7	8
Item 1	0	4	0	0	0	2	1	3
2	3	1	4	3	1	2	0	1
3	3	0	0	0	3	0	3	0
4	0	1	0	3	0	0	2	0
5	2	2	2	3	1	4	0	2

(Table 1) fig: Example of vector

> There should be a measure to calculate the similarity b/w 2 terms.

> The simple measure could be,

$$SIM(Term_i, Term_j) = \Sigma (Term_{k,i}^{\circ})(Term_{k,j}^{\circ})$$

where k is summed across set of all items.

> The effect of the formula would be 2 columns of 2 terms are being analyzed by multiplying & accumulating the values in each row.

> The resu
matrix. T
Term M
> The va
in the
> As the
matrix
> other
non -
> For th
matri

Item 1
2
3
4
5
6
7
8

> The results would be stored in m*m square matrix. That square matrix is known as Term Term matrix.

> The value of m would be the num of columns in the original matrix.

> As the formula is reflexive, the generated matrix would be "Symmetric".

> other similarity formulas could produce a non-symmetric matrix.

> For the data in table 1 the term term matrix would be as shown in below table 2.

	Term 1	2	3	4	5	6	7	8
Item 1		7	16	15	14	14	9	7
2	7		8	12	3	18	6	17
3	16	8		18	6	16	0	8
4	15	12	18		6	18	6	9
5	14	03	6	6		6	9	3
6	14	18	16	18	6		2	16
7	9	6	0	6	9	2		3
8	7	17	8	9	3	16	3	

Table 2 : Term Term Matrix.

> In the above table 2 the diagonal of matrix is empty as it represents the correlation of a word to itself.

> Now, the threshold value should be selected to check whether the 2 values belong to the same cluster or not.

> suppose if the threshold value is 10 for the table 2., the terms with the value of 10 or greater are said to be similar.

The resultant would be "Term relationship matrix" as shown in table 3.

	Term1	2	3	4	5	6	7	8
Item1		0	1	1	1	1	0	0
2	0		0	1	0	1	0	1
3	1	0		1	0	1	0	0
4	1	1	1		0	1	0	0
5	1	0	0	0		0	0	0
6	1	1	1	1	0		0	1
7	0	0	0	0	0	0		0
8	0	1	0	0	0	1	0	

[table] fig: Term Relationship Matrix.

> In the above tab
The value 1 in
term specified
the same class.

> The term 7
clusters as th
term 7.

> The last step
ine the cluster
same cluster.

> There are sev
objects. They
1) cliques 2)
3) stars 4

> Using any
can be solve

> Thus cluste

1) Cliques met

in an cluster
& all other
It can be
1) let i=1
2) select te
3) start with

> In the above table represents a binary matrix. The value 1 in the matrix indicates that the term specified by the column & row can be in the same class.

> The term 7 can't be included in any of the clusters as there are no similar terms for term 7.

> The last step in the process would be to determine the clusters when two objects are in the same cluster.

> There are several methods to determine such objects. They are,

1) cliques 2) single link
3) stars 4) connected components

> Using any of the above methods, the ambiguity can be solved.

> Thus clusters are created.

1) Cliques method:

In cliques method, all items in an cluster should be with in the threshold of all other items.

It can be explained as,

1) let i = 1
2) Select term i and place it in new class.
3) Start with term$_k$ where k = i+1 and r = i+1.

4. Validate if $term_k$ is within the threshold of all terms with in the current class

5. else let k = k+1

6. if k > m (num of words)

then r = r+1

if r = m then go to 7

else

create a new class with $term_i$ in it

go to 4

else goto 4.

7. If the current class has only $term_i$ in it & there are other classes with $term_i$ in them then delete current class

else

i = i+1

8. If i = m+1 then goto 9.

else goto 2

9. Eliminate any classes that duplicate or that are subsets of other classes.

> By applying the above algorithm to the table 3. the clusters that would be created are,

Cluster 1

Cluster 2

Cluster 3

Cluster 4

> The term

one clust

2) Single

Similar to

added to

#Algorithm

Step 1: I

cl

Step 2: Al

clas

Step 3: fo

clas

Step 4: W

g

> A term

clusters

> Applying

can be

cluster 1	{term 1, term 3, 4, 6}
Cluster 2	{term 1, 5}
cluster 3	{term 2, 4, 6}
Cluster 4	{term 2, 6, 8}.

> The term 1 & term 6 are present in more than one cluster.

2) Single Link method:

In this method, any term similar to any term in the cluster can be added to the cluster.

Algorithm

Step 1: Identify the item that is not in the class & place it in new class.

Step 2: All the other terms related to new class must be placed in it.

Step 3: for every term that is entering into the class, perform step 2.

Step 4: When there are no new terms in step 2, goto step 1.

> A term can't exist in 2 or multiple different clusters.

> Applying this algorithm, the matrix in table 3 can be divided as,

cluster 1 {term 1, 2, 3, 4, 5, 6, 8}

cluster 2 {term 7}

3, Star method:

> In this technique, a term is selected & it is placed in a class & all terms related to the term are placed in that class.

> The terms that are not placed in the class are selected & they are placed into classes with the same criteria.

> There can be many different classes created based on star technique.

The clusters of table 3 can be,

Cluster 1 {term 1, term 3, term 4, term 5, term 6}

cluster 2 {term 2, term 4, term 6, term 8}

Cluster 3 {term 7}

> The terms can be in multiple clusters in this method.

This can be eliminated by expanding the constraints to exclude any term that has already been selected for a previous cluster.

4, String method:

> In this method, it starts with the term & the term is included in a cluster & the additional term is added if it is similar to other term if that is not present in the cluster already.

> The new t
process will
terms to be

> The new t
clusters al

> The cluste
in table 3
cluster 1
cluster 2
cluster 3

N/w Diagro

technique
clusters u

> Each term
the nodes

>

t4

> The new term is treated as a new node & the process will be repeated till there are no new terms to be added to the clusters.

> The new term should not exist in any of the clusters already created.

> The clusters that are created for the matrix in table 3 would be,

cluster 1 {term 1, term 3, term 4, term 2, term 6, term 8}

cluster 2 {term 5}

cluster 3 {term 7}

N/w Diagram of the terms

> The n/w diagram is a technique that all the clusters generated by the clusters would be understood easily by this.

> Each term is treated as node and the links b/w the nodes are treated as similar terms.

>

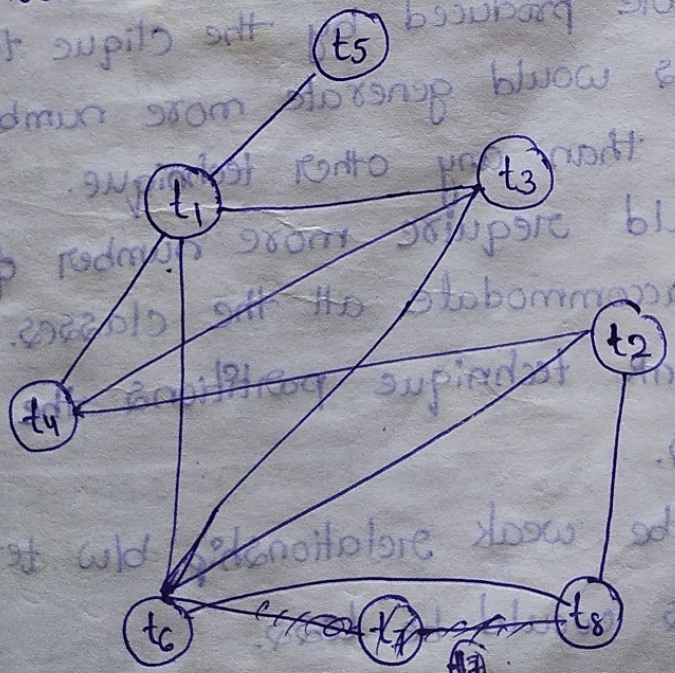


fig: n/w diagram

> To identify cliques, the sub nets are selected where all the items are connected by the links.
> from table 2, term 7 (t_7) is in a class and t_5 is in cluster with t_1.
> The focus should be on triangles & polygons of 4 sides with diagonals.
> To find all the clusters for an item, there is need to find all the subnetworks & there should be maximum num of nodes in each sub n/w.
> For t_1, it is in sub n/w (t_1, t_3, t_4, t_6).
> The term t_2 has 2 sub n/w, one is $\{t_2, t_4, t_6\}$ & other is $\{t_2, t_6, t_8\}$.
> The n/w diagram is a simple visual tool where there would be less num of nodes to identify the clusters.
> The words with the strongest relationships in the cluster are produced by the clique technique.
> This techniques would generate more number of clusters than any other technique.
> Thus it would require more number of clusters to accommodate all the classes.
> The single link technique partitions the terms into clusters.
> There would be weak relationship b/w terms & no of clusters would be less.

> There can
if the valu
cliques pro
> The single li
recall.
> It requires
Clustering
> One more
with the se
> As the proce
there is no
the clusters.
> The terms
are revised
the cluster
> When the
clusters, the

Centroid:

> The centroid
inorder to
> In physics,
known as
centroid is
> The centroid

> There can be 2 terms in the same cluster if the value of similarity is zero.
cliques provide highest precision.
> The single link process would maximize the recall.
> It requires the overhead of $O(n^2)$ comparisons.

Clustering Using Existing Clusters:

> One more method of creating clusters is to start with the set of existing clusters.
> As the process starts with already existing clusters, there is no need to calculate the similar terms in the clusters.
> The terms that are assigned initially to the clusters are revised by validating each & every term in the cluster again.
> When the minimal movement is detected blw the clusters, then the process would be stopped.

Centroid:

> The centroids are calculated for each cluster inorder to minimize the calculations.
> In physics, the centre of mass of set of objects is known as Centroid where as in content of vectors, centroid is the average of all vectors.
> The centroid of the clusters can be viewed as

another point in n dimensional space where n is the number of items.

> The centroids of the initial clusters are not same as the final clusters.

> The relation or similarity b/w all the terms that are existing and the centroids can be calculated.

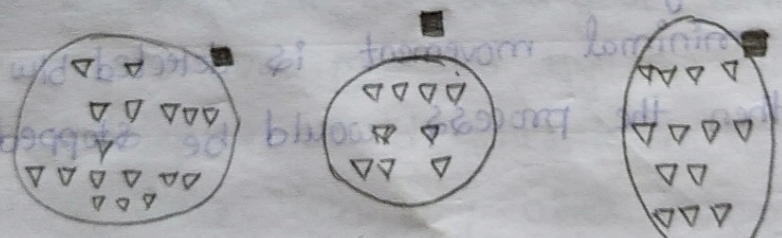
> The term is moved to the cluster with highest similarity. The process will be continued till the cluster stabilizes.

> The order of the calculation in this process is O(n).

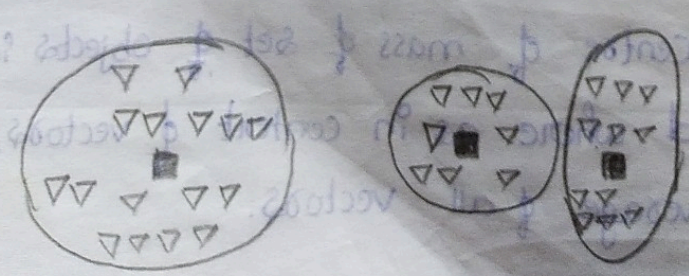
> As the process is iterative & the terms change their clusters till it stabilizes, the initial assignment of terms to clusters is not that important.

Graphical Representation:

The graphical representation of centroids is shown as below



fig(a) initial centroids for clusters.



fig(b) centroids after Repos reassiging terms.

> fig shows initial a

> The solid cluster.

> In fig (a) clusters assignmen

> After p & new are alm

Example:

	ter
Item 1	
2	
3	
4	
5	

It is

Cluste

Cluste

Cluste

The

Cluste

- fig shows how the clusters are moved after initial assignments.
- The solid box represents the centroid for each of cluster.
- In fig (a) represents the centroids of the initial clusters and the ovals represents the ideal cluster assignments.
- After performing the re-assignment, the new centroid & new clusters are represented in fig (b). They are almost similar to the ideal clusters.

Example:

	term1	T2	T3	T4	T5	T6	T7	T8
Item1	0	4	0	0	0	2	1	3
2	3	1	4	3	1	2	0	1
3	3	0	0	0	3	0	3	0
4	0	1	0	3	0	0	2	0
5	2	2	2	3	1	4	0	2

table: vector

It is assumed that the clusters formed are,

Cluster 1 = {term 1, term 2}

Cluster 2 = {term 3, term 4}

Cluster 3 = {term 5, term 6}

The centroids for each cluster would be

$$\text{Cluster 1} = \frac{(0+4)}{2}, \frac{(3+1)}{2}, \frac{(3+0)}{2}, \frac{(0+1)}{2}, \frac{(2+2)}{2}$$

$$= 4/2, 4/2, 3/2, 1/2, 4/2.$$

Cluster 2 = $\frac{(0+0)}{2}, \frac{(4+3)}{2}, \frac{(0+0)}{2}, \frac{(0+3)}{2}, \frac{(2+3)}{2}$

$= 0/2, 7/2, 0/2, 3/2, 5/2$

cluster 3 = $\frac{(0+2)}{2}, \frac{(1+2)}{2}, \frac{(3+0)}{2}, \frac{(0+0)}{2}, \frac{(1+4)}{2}$

$= 2/2, 3/2, 3/2, 0/2, 5/2$

> The value of centroid in each cluster is the avg of the weights of the terms in the cluster.

> The measure of similarity applied here is

$SIM(term_p, term_s) = \Sigma(term_{k,i}) (term_{k,j})$

This method is applied b/w 8 terms & 3 centroids & the result would be as shown in below table.

	Term 1	2	3	4	5	6	7	8
cluster 1	29/2	29/2	24/2	27/2	17/2	32/2	15/2	24/2
2	31/2	20/2	38/2	45/2	12/2	34/2	6/2	17/2
3	28/2	21/2	22/2	24/2	17/2	30/2	11/2	19/2
Assign clusters	2	1	2	2	3	2	1	1

table 5

Fig: Iterated Assignments of clusters.

> In case of similarity.

> This tie weights of

> one that

> most of term is

> In the cluster cluster cluste

> The ments

cluster
2
3
Assign cluste

> By term

> The shown

> In case of term 5, the cluster 1 & cluster 3 have same similarity. 2 clusters can be assigned.
> This tie can be broken by considering the similarity weights of other items in the cluster & assign the one that is having most similar weights.
> most of the terms in cluster 1 are above $2\frac{0}{2}$. so the term is assigned to cluster 3.
> In the same way the next iteration would be,

cluster 1 = $\frac{8}{3}, \frac{2}{3}, \frac{3}{3}, \frac{3}{3}, \frac{4}{3}$

cluster 2 = $\frac{2}{4}, \frac{12}{4}, \frac{3}{4}, \frac{3}{4}, \frac{14}{4}$

cluster 3 = $\frac{0}{1}, \frac{1}{1}, \frac{3}{1}, \frac{0}{1}, \frac{1}{1}$

Iteration

class 1 = (2, 7, 8)

2 = (1, 3, 4, 6)

3 = (5)

> The above are new centroids & the cluster assignments would be as shown in table below.

	Term 1	2	3	4	5	6	7	8
cluster 1	23/3	45/3	16/3	27/3	15/3	36/3	23/3	34/3
2	67/4	45/4	70/4	78/4	33/4	72/4	17/4	40/4
3	12/1	3/1	6/1	6/1	11/1	6/1	9/1	3/1
Assign clusters	2	1	2	2	3	2	3	1

table:6: Cluster Assignment

> By comparing table 5 & 6, the only change is term 1 is changed from cluster 1 to cluster 3.
> The reason would be the term 7 is not strongly related to cluster 1.

Limitation:

> Although there are less calculations in this method when compared to complete Term Relationship method, this method has few limitations.

> The num of clusters are defined at the starting of the process & they can't be expanded or increased.

> It is also possible that there would be less number of clusters at the end of the process.

> even if the similarity is relatively weaker, the term would be assigned to that cluster because all the items should be assigned to some or other clusters.

One Pass Assignments:

> This technique of clustering has minimum overhead because one pass of all the terms is used to assign the terms to the clusters.

> In this technique, the 1st term is assigned to the 1st cluster.

> The additional terms are compared with the centroids of the existing clusters. A threshold value is choosen & if the item has greater weight than the threshold, then it is assigned with the highest similarity one.

> For the clu
of centroid

> If the simil
less than the
item in a

Example:

> Consider
table. & le

	Terms
Items	
2	7
3	16
4	15
5	14
6	14
7	9
8	7

the clus

Cluster 1

Cluster 2

Cluster 3

Cluster

> for the cluster that is modified, a new value of centroid have to be found.

> If the similarity of all the centroids existing is less than that of threshold, then item is the 1st item in a new class.

Example:

> Consider the matrix that is represented in below table. & let the threshold value be 10, then

	Terms	2	3	4	5	6	7	8
Items		7	16	15	14	14	9	7
2	7		8	12	13	18	6	17
3	16	8		18	6	16	0	8
4	15	12	18		6	18	6	9
5	14	3	6	6		6	9	3
6	14	18	16	18	6		2	16
7	9	6	0	6	9	2		3
8	7	17	8	9	3	16	3	

table: term term matrix.

the clusters are,

Cluster 1 = {term 1, term 3, term 4}

Cluster 2 = {term 2, term 6, term 8}

Cluster 3 = {term 5}

Cluster 4 = {term 7}.

> During one pass process, the centroids values used are,

cluster 1 {term 1, 3} = 0, 7/2, 3/2, 0, 4/2

cluster 1 {term 1, 3, 4} = 0, 10/3, 3/3, 3/3, 7/3

cluster 2 {term 2, 6} = 6/2, 3/2, 0/2, 1/2, 6/2.

Limitations:

> Eventhough, this process has less computational overhead of order O(n), it can produce optimal results i.e., optimum clustered classes or clusters.

> If the order of the items in analysis are changed then different clusters would be produced.

> Due to the averaging nature of centroids, the items in same cluster would be appeared as they are in different clusters.

Item clustering:

> In the sawfi generation, the item clustering & term clustering are almost similar.

> Any library or filling system uses manual item clustering.

> In manual clustering, one person would read the item & determine the category into which the item belongs.

> The item
category in
> But with
would be
category &
categories
assigned
> Because
automati
developed
> the me
be used
> The
i, compl
ii, Clust
iii, One
> The
on the
Vs ter
> The
Colum
> The

> The item is generally assigned to a single category in physical clustering.

> But with the introduction of indexing, the item would be stored physically in the primary category & the item is found in all the other categories that are defined by index items assigned to it.

> Because all the items are stored electronically, the automatic clustering techniques have also been developed.

> the methods of automatic term clustering can also be used for clustering the items.

> The methods include,

(i) complete term Relation method.

(ii) clustering using existing clusters.

(iii) One Pass Assignments.

> The similarity b/w the documents can be known on the basis of 2 items having the common terms V/s terms with items in common.

> The similarity fun^n is performed b/w rows & columns of the item matrix.

> The measure of similarity would be calculated by,

$$SIM(item_i, item_j) = \Sigma (term_{i,k})(term_{j,k})$$

Example: Consider the vector table,

	Term 1	2	3	4	5	6	7	8
Item1	0	4	0	0	0	2	1	3
2	3	1	4	3	1	2	0	1
3	3	0	0	0	3	0	3	0
4	0	1	0	3	0	0	2	0
5	2	2	2	3	1	4	0	2

the values of k goes from 1 to 8, the item-item matrix would be as shown in below table.

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

table: Item - item matrix

> If the threshold value is 10, then the item relationship matrix could be as shown in below table.

	Item
Item 1	
2	
3	
4	
5	

> The clus
above
cluster
Cluste
Cluste

> using
Created
cluste

> The St
cluster

cluste
Cluste
Cluste

> The

Clust
Clust

	Item 1	2	3	4	5
Item 1		1	0	0	1
2	1		1	1	1
3	0	1		0	0
4	0	1	0		1
5	1	1	0	1	

table: Item Relationship matrix.

> The clusters using cliques technique would be for above table.

cluster 1 = {item 1, item 2, item 5}

Cluster 2 = {item 2, item 3}

Cluster 3 = {item 2, 4, 5}

> using Single link technique, the clusters created would be,

cluster 1 = {item 1, 2, 3, 4, 5}

> The star technique would generate the following clusters.

cluster 1 = {item 1, 2, 5}

Cluster 2 = {item 3, 2}

Cluster 3 = {item 4, 2, 5}

> The string technique would generate

Cluster 1 = {item 1, 2, 3}

Cluster 2 = {item 4, 5}

> The ambiguities & erroneous hits are caused by vocabulary domain homographs.

> The process of clustering can also be started with the existing clusters.

> If,

cluster 1 = { item 1, item 3 }

cluster 2 = { item 2, item 4 }

then centroids are,

cluster 1 = 3/2, 4/2, 0/2, 0/2, 3/2, 2/2, 4/2, 3/2

cluster 2 = 3/2, 2/2, 4/2, 6/2, 1/2, 2/2, 2/2, 1/2

> After calculating again the results would be as in below table

	Item 1	2	3	4	5
cluster 1	33/2	23/2	30/2	8/2	31/2
cluster 2	17/2	51/2	18/2	84/2	47/2
Assign clusters	1	2	2	2	2

table: clustering of items with existing clusters

> Now the cluster 2 contains 4 items and cluster 1 only one item. So recalculating does not result in any re-assignment.

Hierarchy of Clusters:

> In Information Retrieval, the hierarchical clustering concentrates mainly on the concept of hierarchical agglomerative clustering methods (HACM)

> In agglomerative clustering, the process would start with unclustered items & then similarity measures are performed pair wise to determine the clusters.

Objectives:

The objectives of hierarchy of clusters would be,

(i) Reduction of search overhead.

(ii) Visual representation of information space.

(iii) Relevant item retrieval expansion.

(i) Reduction of search overhead:

The overhead in searching is reduced by performing top down searches of the centroids of clusters present in the hierarchy & by trimming (cut) the branches of hierarchy that are not relevant.

(ii) Visual representation of information space:

> It would be very difficult task to display the complete item space visually.

> The pictorial representations known as "Dendograms" would help the user to determine the clusters to

be reviewed & likely to find the relevant items.

> The size of clusters is defined by size of ellipse in dendograms.

> The linkage strengths are indicated by lines. Dashed lines indicated reduced similarity.

> It allows the user to brow the alternate paths to database.

> The example of dendogram is as shown in below

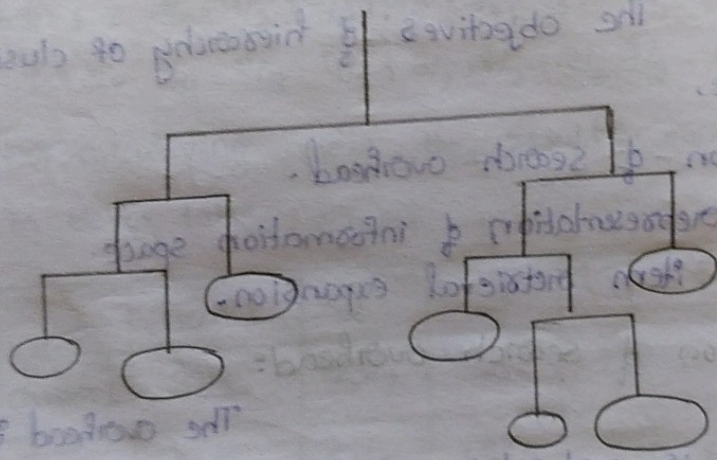


fig: Dendogram

(iii) Relevant Items Retrieval:

> The items that are relevant can be browsed by using logical hierarchy instead of visual display hierarchy.

> Once the relevant item is identified, the user can send the request to other items in the cluster.

> The specification of the item can be increased by passing through child clusters by increasing the generality by reviewing the parent clusters.

—o—