Ut 3: Evaluation and Visualisation of ISR.

* **MRR :-** 1) The mean reciprocal rank - MRR is a statistical metric for accessing any produre that generates a list of potential answer to a sample question, ordered by likelihood of correctness.

2) The reciprocal rank of a query response is the mutiplicative inverse of rank of first correct answer : 1 for first place, 1/2 for second, 1/3 for third place & so on.

3) The mean reciprocal rank is average of the reciprocal ranks of results for a sample of queries.

4) Mathematically MRR can be represented as,

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}$$

where $rank_i$ refers to the rank position of first relevant document for ith query.

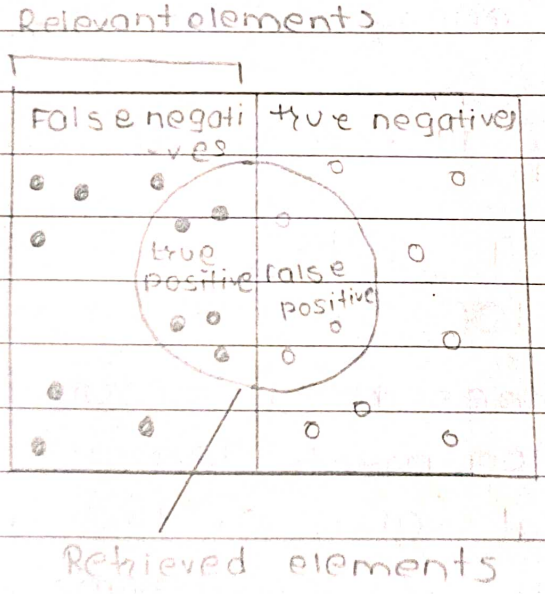
5) The reciprocal value of the mean reciproca rank corresponds to the harmonic mean of ranks.

* **F-SCORE :-**

1) In statistical analysic of binary classification, the F-score or f-measure is a measure of a test's accuracy.

2) It is calculated from the precision & recall of the test, where the precision is the number of true positive results divided by the no. of all positive results, including those not identified correctly, & the recall is the no. of true positive results divided by the no. of all samples that should have been identified as positive.

3) Precision is also known as positive predictive value & recall is known as sensitivity in diagnostic binary classification.



4) The F_1 score is harmonic mean of precision & recall. The more generic F_β score applies additional weights, value one of precision or recall more than other.

5) The highest possible value of an F-score is 1.0 & lowest possible value is 0.

6) F-score is the harmonic mean of precision & recall:

$$F_1 = \frac{2}{recall^{-1} + precision^{-1}} = 2\frac{Precsion \cdot recall}{Precsion + recall}$$

$$= \frac{tp}{tp + \frac{1}{2}(fp+fn)}$$

f_β score,

A more general F score, f_β, that uses a positive real factor β, where β is chosen such that recall is considered β times as important as precision is :-

$$F_\beta = (1+\beta^2) \cdot \frac{Precsion \cdot recall}{(\beta^2 \cdot precision) + recall}$$

* NDCG :-

- Cumulative Gain :-

1) The graded relevance values of each result in a search result list are added up to form cumulative Gain (CG).
2) The rank of a result in the result list is not taken into account when determining the usefulness of a result set by this DCG ancestor.
3) CG at rank position p :

$$CG_p = \sum_{i=1}^{p} rel_i$$

where rel_i is the result's graded relevance at position i.

4) changes in the order of the search results have no impact on value calculated using the CG function.

5) As it is same as Precision metric if the rating scale is binary, CG is sometimes referred to as Graded Precision.

* Discounted Cumulative Gain :-

1) Ranking quality is measured by discounted cumulative gain (DCG)

2) It is frequently used in information retrievel to gauge the efficiency of algorithm employed in Web search engines.

3) DCG calculates the utility, or gain of a content based on its placement in result list using a graded relevance scale of items in a search engine result set.

4) In using DCG & its associated measurement two assumption are made.
   1. Documents that are highly relevant are more helful when they appear earlier in a search engine result list.
   2. Papers with a high degree of relevance are more helful than those with a low degree of relevance, which in turn are more helpful than documents without a high

degree of relevance.

5. DCG accumulated at a specific rank position p:-

$$DCG_p = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log(i+1)}$$

6. DCG places stronger emphasis on retrieving relevant documents:

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2(i+1)}$$

* Normalized DCG:-

- nDCG, is calculated for a query as:

$$nDCG_p = \frac{DCG_p}{IDCG_p}$$

where ideal discounted cumulative gain (IDCG) exists.

$$IDCG = \sum_{i=1}^{|REI_p|} \frac{rel_i}{\log_2(i+1)}$$

The fundamental issue with employing nDCG is that a perfect ordering of result is not possible when just partial relevance feedback is supplies.

* USER-ORIENTED MEASURES —

1) Let A represent the set of retrived ans, & let R represent the collection of pertinet document for I
2) U be subset of R which known as user.
3) The number of documents is U is |U|.
4) |Rk) be number of documents in set.

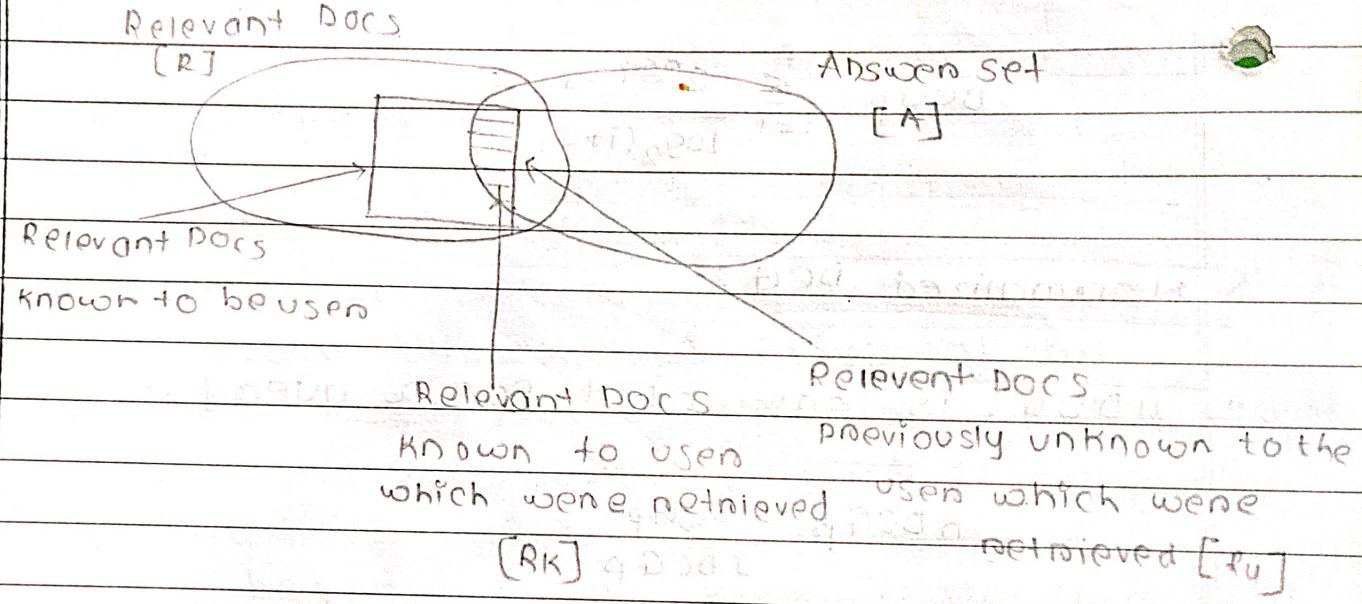


Fig. User Oriented Measures.

- $\text{Cov range} = \frac{|R_K|}{U}$
- $\text{novelty} = \frac{|R_U|}{|R_U| + |R_K|}$

MRR :- The mean reciprocal rank - MRR

* Visualization in Information System :- Starting Points.

1) Users must have easy entry points in search interfaces.
2) Users typically don't start by expressing their information needs in great detail.
3) Users frequently begin their inquiries with extremely brief phases
4) For a sense of rephrase the query, the initial query can be seen of as a kind of "testing the water".

- Lists of collections :-

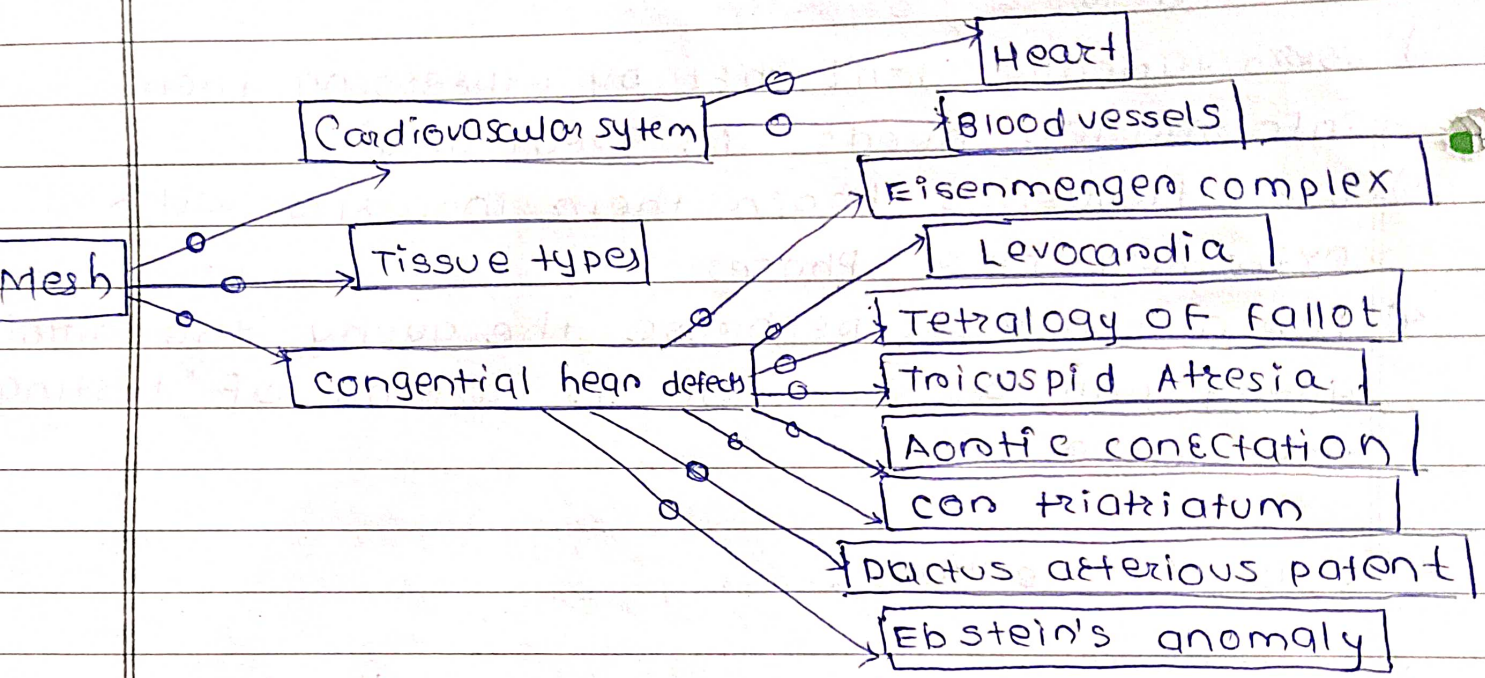
1) Users of standard online system like LEXIS-NEXIS are required to start any enquiry by scanning through a vast list of source names.
2) A new query must be issued on a different collection if the user is dissatisfied with results on one collection

- Overviews :-

1) To assist user in choosing or excluding source from consideration, an overview may display the topic domains represented within collections

2) An interaction model where the user start with an overview of data they'll be working with, then pans & zooms to locate possible interest ares & the views details.

- Category or Directory Overviews :-



* Query Specification

In order to create a query, a user must first choose the collections, metadata descriptions, on information sets that the query will be compared to. & then they must provide the words, phrases, descriptors or other types of data that will be compared to or matched against the collection's data.

1) Boolean Queries :-

- Modern information access systems typically use a statistical ranking algorithm for their matching procedure.
- Boolean inquiries are challenging for a no. of reasons.
- The primary one is that most people find the fundamental syntax couter-intuitive.

2) Faceted Queries :-

- Another issue with Boolean searches is that, when strictly interpreted, they frequently produce result sets or contains numerous terms in a disjuct.
- The user's ignorance about the collection's contents & the fun of its

* User Relevance Judgement.

- Reformulating queries is an essential step in info. access process, and relevance feedback is a tried and true method for doing so.
- In its original form, relevance feedback referred to a cycle of user-system interaction in which the user chose a seled group of doc. they believed to be pertinent to query, and the system subsequently used features generated from these chosen document to modify the org. question.
- Relevance feedback introduces crucial design decisions, such as which actions should be taken by the system automatically & which should be taken by the user and controlled by them.
- It's often best to use a quick, simple technique that doesn't involve much knowledge or consideration.
- Sometimes a user require more control over a piece of equipment's operation, but they still don't want to know the specific of how it work on a basic level.
- features that significantly simplify the formulation of intial queries can be added to these simple user interfaces.
- A query preview may provide this info. as the user types the query as opposed to after the user has executed the query have recieved

* Interface support for search process.

① Interface for string matching

- A typical easy search need is to utilise the "find" operation to search through the contents of a document that is now being shown.
- The majority of the time, this method just executes a simple string match & does not produce ranking o/p or support Boolean phrase combinations.
- In most case, a specialised search window is constructed with a few simple buttons.
- The next level of sophistication is the "find" capability for searching through smaller collections, such as the' datb.
- The next level of sophistication is the "find" capability for searching through smaller collections, such as the data on computer's hard driven or history list of a web browser. This kind of function is usually implemented as a plain string match.
- The controls and parameter setting for a special purpose search window are once more visible at the top & various parameters are supplied using checkboxes & entry forms.
- Features that significantly simplify the formulation of intial queries can be added to these simple user interfaces.

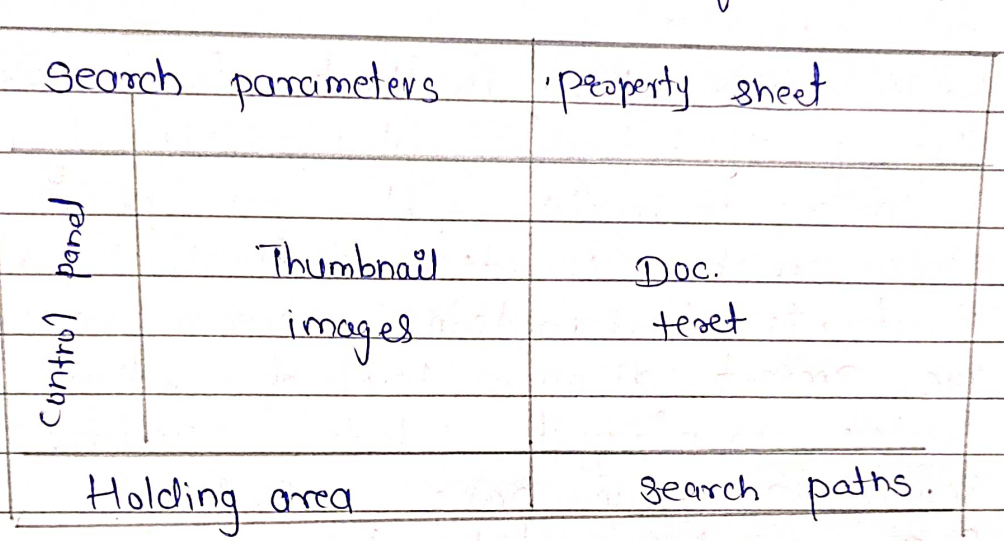
(2) Window Management

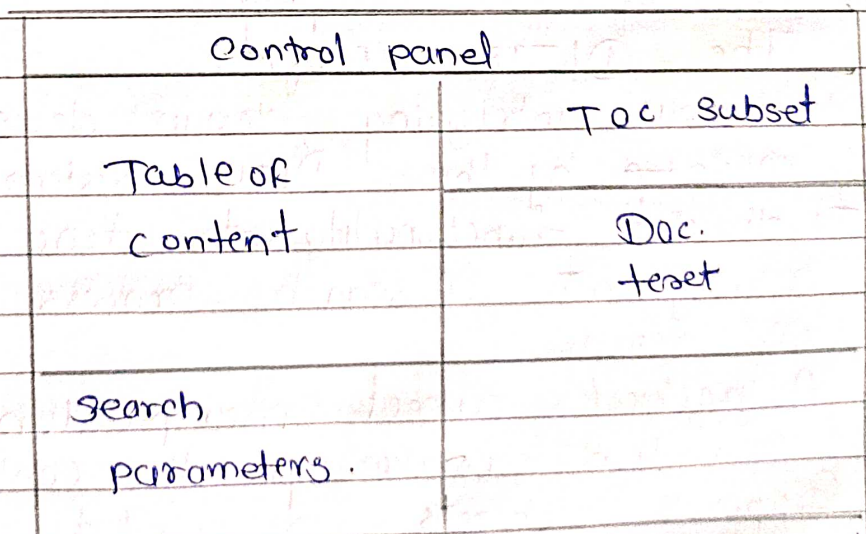
- For search tasks that are more complex than the fundamental string matching find procedures listed above, the interface designer must decide how to arrange multiple options & info. displays within interface.
- Linking the information in one window to info. in another window is freq. helpful in info. access systems.
- When arranging info. within windows, the designer must choosen betn a monolithic display, in which all the windows are laid out in predefined positions & are all simultaneously viewable, tiled windows, and overlapping windows.
- Many benefits come with monolithic interface. The elements are placed in recognisable locations, making them easier to identify, & the designer has complete freedom over how options are arranged. But there are other drawbacks to monolithic interfaces.
- The user can resize a full group of windows with a single gesture, which automatically reduce the size of the other workspaces to fit everything on the screen without overlapping.

* Example Systems.

① The infogrid layout

- Infogrid system is a prime illustration of a monolithic interface design for information access.
- A huge display must be available for the design, which is seprated into a left & right side.





- The left side is further divided into a top portion with structured entry forms for

describing the properties of a query, a colaumn of recognisable controls listing the left side. and a bottom space for keeping crucial paperes.

The Superbook Layout.

The InfoGrid's design is remarkably similar to superbook's

The key distinction is that Superebook's the display that function like a table of content in main left-hand pane, along with counteres that show no. of documents that contain search reseult that are present at each level of outline.

Similar, to info grid, selected documents are shown in main pane on right side screen.

Query creation takes plaie just beneath table of content display.

The DLITE interface.

Numerous intriguing design decision are made ~~up~~ by the DLITE system.

It divides functionality into two sections : management of search process and display of results.

A graphical direct manipulation display with animation makes up the control section

Graphical objects are used to represent queeris, sources, documents and collections of retrived documents.

- The system creates a query object, which is represented by a tiny icon & may be put onto icons for collections or search services.
- Documents can be put into other services such a document summarize or a language translator, after being pulled out results set pool.

④ The SketchTrieve Interface

- The SketchTrieve interface is based on the idea that information access should be portrayed as an informal process, where incomplete concepts & partially explored avenues can be saved and returned to compare to subsequent interactions, and where results ~~are~~ can be combined using operations on graphical objects & connectors between them.
- In future, there should be a wider adoption of idea of a canvas or workspace for the preservation of previous context.
- Numerous problems are difficult to resolve such as how to display outcomes of a collection of linked searches after making tiny alterations based on query expansion, relevance feedback, & other types of modification.