Enhanced Performance of Search Engine with Multi-Type Feature Co-Selection for Clustering Algorithm

K. Parimala
Assistant Professor,
MCA Department,
NMS S.Vellaichamy Nadar College,
Madurai.

V. Palanisamy, Ph.D
Associate Professor,
Department of Computer Science & Engineering,
Alagappa University,
Karaikudi.

ABSTRACT

Information retrieval is a science of gathering information from unstructured data, the online information source i.e., www. WWW contains data of heterogeneous types and of high dimension. Retrieving information from such database is a tedious work. Many researches are going on, to find a best optimal solution. A search engine is the tool for retrieving information from www. The internet helps the user to get the required information from www. A search engine respond to the user-need by answering their query, contains: Database, Web crawler, and Ranking algorithm. The optimality of the search engine is based on the ranking algorithm. The rank list is prepared based on the relevancy score. In this work we propose to use a novel algorithm, Multi-type Feature Co-selection for Clustering (MFCC) to the search engine as an alternative for the ranking algorithm. MFCC has proved its efficiency in clustering the heterogeneous web documentation.

Keywords: Information Retrieval, Search Engine, MFCC algorithm, Feature selection.

1. INTRODUCTION

“Information Retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)” [1]

IR used to be an activity that only a few people engaged in reference libraries, paralegals and similar professional searchers. Now the world has changed and hundreds of millions of people engage in IR everyday when they use a web search engine. IR is fast becoming the dominant form of information access, overtaking traditional database style searching.

IR also covers other kinds of data & information problems. The term “unstructured data” refers to data which does not have clear, semantically overt, easy for a computer structure; it supports the users in browsing or filtering document collections or further processing a set of retrieved documents, grouping or clustering of the documents based on their contents; it distinguishes the scale at which they operate Enterprise, Institutional and domain-specific search, where retrieval might be provided for collections such as Corporation’s Internal Documents, A Database of Patents or Research Articles.

IR focuses on retrieving documents based on the content of their unstructured components. An IR request (query) may specify desired characteristics of both structured and unstructured components of the documents to be retrieved. It typically seeks to find documents in a given collection that are ‘about’ a given topic or that satisfy given information need. Documents that satisfy the given query in the judgment of the user are said to ‘relevant’. It is shown in fig-1.

IR is an academic discipline, which underlies computer based text search tools. It tends to concentrate on mathematical models and algorithms for retrieval quality. It begins with user query, formal statement of information need; it does not uniquely identify a single object in the collection. Instead, several objects may match a query (measured with similarity measure sim(q, di) where q is query d_i is the document collection, $1 \leq i \leq n$).

Inverse Document Frequency (IDF) assumes that the importance of a query (keyword) in calculating similarity measures is inversely proportional to the number of documents that contain it. Given a query, Q and n documents,

$$IDF_q = \log\left(\frac{n}{documents\ containing\ Q}\right) + 1$$

Two measures of IR success, both based on the concept of relevance: Precision (measure of exactness) and Recall (measure of completeness).

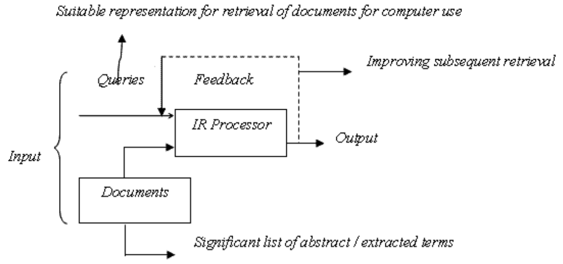

Suitable representation for retrieval of documents for computer use
Queries
Feedback
Improving subsequent retrieval
IR Processor
Output
Input
Documents
Significant list of abstract / extracted terms

Fig –1 A typical IR System

IR is formulated into following mapping in modern Algebra as

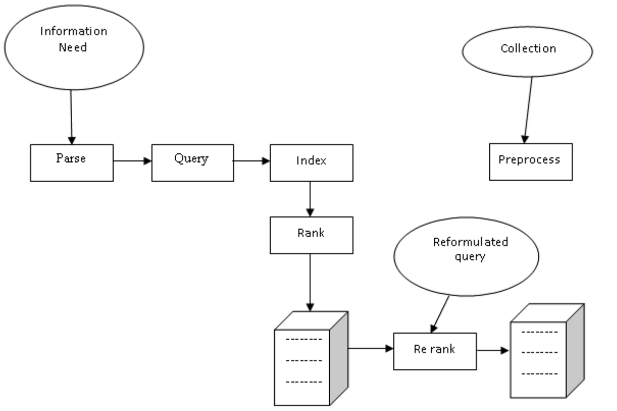
IR: (U, IN, Q, O) $\rightarrow$ R

U: User
IN (Q, 1): information need
Q: Query – in the language of the user.
O: collection of objects to be searched.
R: Collection of retrieval objects in
response to Q (relevance relationship).

IR = R (O, IN) = R (O, (Q, 1)) = R (O, Q, $< I, \rightarrow >$) where $< I, \rightarrow >$ represents the information to be inferred.

Most IR systems computes a numeric score on how well each object in the database match the queries and rank the objects according to its value. The top ranking objects are shown to the user. Process may then be iterated if user wishes to refine the query.

IR research is mainly focused on the retrieval of natural language text in the voluminous textual data spread widely in the internet and also on the private archives [2].



Fig–2 Information Retrieval in Search Engine

A program that searches documents for specified keywords and returns a list of the documents where the keywords were found is Search Engine (see fig-2). Although search engines are one of the best methods to retrieve information from World Wide Web, most popular search engines retrieve documents that match the user specified query terms.

2. SEARCH ENGINE

The web creates new challenges for IR. The amount of information in the web is growing rapidly, as well as the number of new users inexperienced in the art of web search. People like to surf the web using its link graph, often starting with high quality human maintained indices with search engines.

[5] The search engines are classified into three types: robot based search engines; directory based search engines and Meta search engines according to the information collection centre.

i) Robot based search engines traversal web in a certain strategy using s/w robot, download web documents and buildup a huge-scale index. Upon receiving a query, they retrieve the index database and return results relevant to the query.

ii) Directories based search engines collect web information by artificial collection or website authors' initiative commitment, and organize resources in tree-structured directories classified by subject.

iii) Meta search engines based on their services of several individual search engines. They borrow services provided by their member search engines and return the integrated results. They neither own an index database nor a classification directory, which is the biggest difference with individual search engines.

Search engine technology has had to scale dramatically to keep up with the growth of the web. For engineers, search engine is a challenging task. Search engines index web pages involving a comparable number of district terms. They answer queries every day.

The most important challenge for web searching is getting users the information what they seek and it's all about user- experienced relevance.

To quickly extract specific, relevant information from the internet, the serious Searcher must be familiar with the Structure, Functionality, Strengths, Weakness and special features of the most efficient search engines

i) Index based search engines. <u>Ex.</u> www.about.com

ii) Free text search engines. <u>Ex.</u>www.yahoo.com, Attavista.com, Google.com, hotbot.com

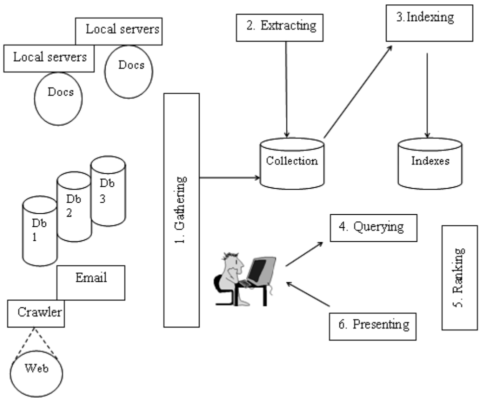
iii) Specialty search engines <u>Ex.</u> www.biolinks.com, searchpdf.adob.com, www.askjeevas.com

iv) Meta sites – specialized directories related to a particular topic purpose is to direct you to other sites on the web. <u>Ex.</u>www.clearinghouse.net, www.bjpinchbeck.com

v) Intelligent agents – desktop portals, desktop browser search tools, browsing companions. <u>Ex.</u>www.copernic.com,http://info.intelliseek.com/prod/bullseys.htm

Search engines change continuously. To keep on top of these changes, we have to follow the technology it follows and methodology of it.

Each search engine works as the division it belongs to, the infrastructure of search engine use, crawling, indexes, spamming, and hashing function.



Fig–3 Search Engine Architecture

The functionality of search engine mainly depends on the indexes, ranking and its presentation (fig-3). Each search engine distinguishes themselves by having individual technology in ranking and indexing, such that their workability also differs. The workability of the search engine can be viewed or testing by feeding same keyword to different search engines.

The result is mostly in the favour of free text search engines [9]. But problem behind this is the search engine generally crawls in the pages it frequently or recently visited. So the fresh or most recently uploaded pages lose its reliability even though it is the most relevant document according to the keyword. Han & et al., has proposed new algorithm for better crawling so that they included new pages also for indexing, such that recently loaded relevant pages are also taken for ranking.

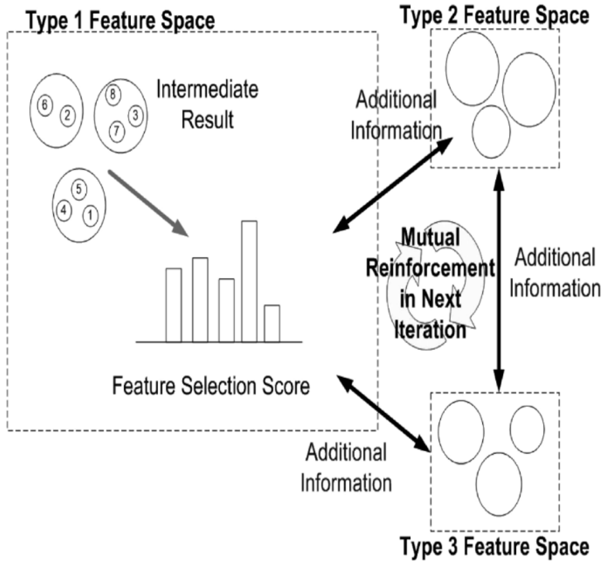
The www, internet database has indexed the documents or web pages under classification and query processed of the internet, the search engine takes them accordingly for the ranking. Certain classification stands outside the ranking because of the categorization even though they are relevant. Feature selection can be used for classification or clustering the web documents, such that information retrieval will be easy because of the dimension reduction and the preprocessing techniques it adopts. A novel algorithm MFCC have proved itself best in clustering web documents pseudo class a new approach in removing classification and iterative feature selection will take outliers also for processing so that all documents can be taken for consideration. The outliers are reduced in clustering web documents in MFCC algorithm.

3. MFCC

First, it should made clear that the selection of each type feature and the clustering is an iterative one. After the iteration of clustering, data object will be assigned to a cluster, each cluster is assumed to correspond for a real class [12]. Using this information, supervised feature selection such as Information Gain (IG) and χ2 statistic (CHI) [13] during k-means clustering is done. MFCC tries to fully exploit heterogeneous features of a web page like URL, anchor text, hyperlink, etc., and to find more discriminative features for unsupervised learning.

We first use different types of features to do clustering independently. Then, we get different sets of pseudo class, which are all used to conduct iterative feature selection (IF) for each feature space.

After normal selection, some data fusion methods are used to conduct iterative feature selection (IF) for each feature space, i.e., feature coselection. In the iteration of clustering, the coselections in several spaces are conducted one by one after clustering results in different feature spaces have been achieved before any coselection. Thus, the sequence of coselection will not affect the final performance. The general idea of coselection for k-means clustering is described in Figure.4.



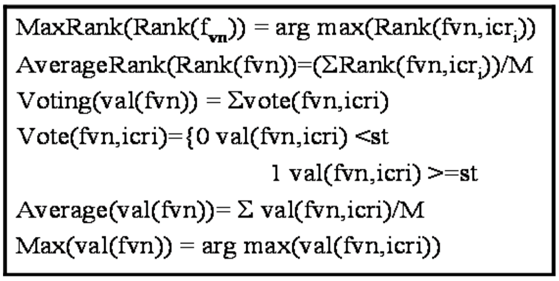
Fig–4 Basic idea of MFCC

Suppose that we categorize data objects with M heterogeneous features into L clusters. Let fv_n be one dimension of the feature vector, icr_i be the intermediate clustering results in the i^{th} feature space, SF be the fusion function. The pseudo algorithm is listed as follows:

```
Loop for N iterations of k-means clustering
  {
    Loop for m feature spaces
      {
          Do clustering in feature space m
      }
    Loop for M feature spaces
    {
          For feature space m, do feature selection using results in all feature spaces.
          For $(fv_n)$ one dimension of the feature vector in space m, a feature selection
          Score fss $(fv_n, icr_i)$ is obtained by using intermediate clustering results $icr_i$ in feature space i.
          Then a combined score fss ( $fv_n$ ) is achieved by fusing the scores based on different result sets.
              $fss(fv_n) = SF(fss(fv_n, icr_i))$     - (1)
          }
    }
```

In the equation (1), (Fss $\left(F\, v_n \right)_{i=1}^{m}$, icr_i) can be the value calculated by the selection function or rank among all features. The feature selection criteria, the six commonly used feature selection function.

Depending on the choices of fss and SF, we obtain five fusion models including voting, average value, max value, average rank, and max rank. The equations are listed as follows:



MaxRank(Rank(f_{vn})) = arg max(Rank(fvn,icr_i))
AverageRank(Rank(fvn))=(ΣRank(fvn,icr_i))/M
Voting(val(fvn)) = Σvote(fvn,icri)
Vote(fvn,icri)={0 val(fvn,icri) <st
1 val(fvn,icri) >=st
Average(val(fvn))= Σ val(fvn,icri)/M
Max(val(fvn)) = arg max(val(fvn,icri))

In the above equation, *val(fv_n,icr_i)* is the value calculated by selection function, *RANK(fv_n,icr_i)* is the rank of fv_n in the whole feature list ordered by *val(fv_n,icr_i)*, and *st* is the threshold of feature selection. After feature coselection, objects will be reassigned, features will be reselected, and the pseudoclass-based selection score will be recombined in the next iteration. Finally, the iterative clustering and feature coselection are well integrated.

In each of the iterations, the whole feature space should be reconsidered. The reason is that our method can help in finding more effective features through a mutual reinforcement process. Properly selected features will help clustering and vice-versa. That is to say, some discriminative features will not be found until late in the clustering phase. This is proved by empirical results.

4. PROPOSED WORK

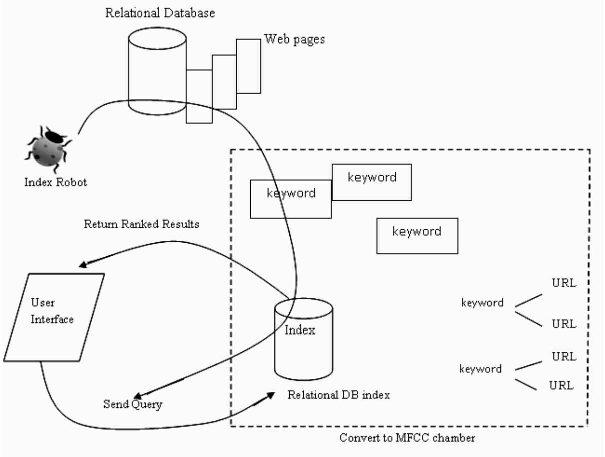
In the proposed work the most common web challenges are focused: spam, content quality, quality evaluation, web convention, duplicate hosts and vaguely structured data. As Pseudoclass were introduced the class identifier such as text, structure, utility, etc. are removed and clusters into feature spaces. Iterative feature clustering helps to remove outliers, so that the problem of fresh or new web pages in search results is also solved.

MFCC has proved its clustering efficiency in web documentation [10] for the databases: www.opendirectory, www.project.com. The result shows that the clustering features have better relevancy than any other. Also it has provided its integrity in text classifiers also [11], [12].

This MFCC is better than the ranking algorithm. Since ranking algorithm, prepares the rank list based on the relevancy score. Then links are matched according to the citations and grouped. But in MFCC it groups or classifies the dataset in to feature spaces. In that, the feature selection score (fss) the best information is selected (SF) from each feature space. This is clustered iteratively.

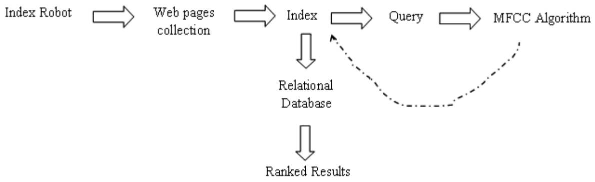
MFCC trains the noisy data and uses that also for the score, no such help form ranking algorithm. Such consistency can be implemented in search engine technology to improve its ranking results.

The proposed architecture is likely to implement in the database index shown in Fig-6.



Fig–6 Search Engine Model

The proposed architecture generates better ranking results, since MFCC does double verification. It clusters according to features; there best distance formulae were implemented to produce quality clusters.



Fig–7 Data Flow of Search Engine Architecture

The iterative feature cluster removes feature class to pseudo class and feature co-selection is implemented so the web pages that are relevant but related to the query is considered. Thus outliers are reduced. In search engine technology, the outliers may be the non relevant web pages but related to keyword, the newly uploaded web pages. In due, this frequent refreshment of index database for crawler algorithm is reduced. Since VSM of all documents are considered for search, and those fresh pages, non relevant but related documents are also taken. The TDM of documents remove all classifiers and consider only TF – IDF, so that feature space is refined.

If the MFCC algorithm is implemented in search engine technology- the rank list are improved. Time precision also can be improved and maintained than any other clustering, classification, machine learning techniques.

5. CONCLUSION

MFCC exploits the different types of feature classes to perform web document clustering. This has been implemented in search engine technology to improve the rank results. The coselection among other feature space, and intermediate clustering results in fusion function. So that the database index is fresh and always takes the entire web pages for clustering. Finally, the usage of MFCC in IR searching architecture reduces the noisy data. The future scope of this architecture frame is put to test and continued for other data sets than textual.

6. REFERENCES

[1] Christopher D. Meaning & et al.; "An introduction to information retrieval"; Cambridge University Press, 2009.

[2] Ed. Green grass, "Information Retrieval: A survey"; 2000.

[3] Sew Staff, "How search engines work", 2007.

[4] Sergey Brie & Lawrence Page, "The Anatomy of a large-scale hyper textual web search engine" 2009.

[5] Lin Guoyuan & et al., "Studies & evaluation on Meta search engines", 2011 IEEE, 978-1-61284-840-2/11.

[6] Joseph Williams and Ravi Starzi, "Tuning up the search engine", IT-PRO Jan/106-2011, 15 20-9202/01/2001 IEEE.

[7] Kristen L.Metzger, "Advanced web searching for the information professional".

[8] David Hawking, "Web search engines: part 1 & part 2", CSIRO/CT centre 2006; pg.86-89, June 2006; "Computer: How things work" pg.88489, Aug. 2006.

[9] K.Parimala, & V.Palanichamy, "A comparative study on search engines in information retrieval", ICSTAOR IT – 2006, XXVI ISPS Conference proceedings.

[10] Han & et al., "Multi type feature co-selection for clustering for web documentation", IEEE transaction on knowledge engineering, June 2006.

[11] Srinivas M & et al., "MFCC and ARM algorithms for text categorization", Aug 2010.

[12] Srinivas M & et al., "Improving performance of Text categorization: Using MFCC and LSquare Machine Learning", 2010.

[13] Y. Yang and J.O. Pedersen, "A Comparative Study on feature Selection in Text categorization," Proc. Int'l Conf. Machine Learning (ICML '97), pp. 412-420, 1997.