ISSN No. 0976-5697

Volume 5, No. 2, March 2014 (Special Issue)

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Search Results Clustering Validation Techniques

K.Hima Bindu
Dept of CSE
Vishnu Institute of Technology
Bhimavaram, India
himagopal@gmail.com

R. Srnivasa Raju
Dept of CSE
Vishnu Institute of Technology
Bhimavaram, India
r.srinivasaraju@ gmail.com

D. Surya Narayana
Dept of CSE
Vishnu Institute of Technology
Bhimavaram, India
surya_dasika@yahoo.com

Abstract: Search Results Clustering (SRC) is a solution for information abundance caused due to ambiguous web search queries. SRC has its own unique challenges in contrast to classical clustering techniques. In this paper, we describe the fundamental concepts of SRC while surveying the quality assessment measures.

Keywords: Information retrieval, Meta search engines, Text clustering, Search results clustering, Web mining

I. INTRODUCTION

The flat ranked list of search results returned by a search engine usually contains millions of results in case of short and ambiguous queries. However, checking the long list of results is a tedious experience. Users usually check the first few pages of the results to find the relevant results. It is reported that 3% of web queries and 23% of most frequent queries are ambiguous [37]. Search Results Clustering (SRC) is a well-known approach to handle the lexical ambiguity raised due to short and ambiguous queries [3]. It identifies the topics or categories related to the query, by clustering the short text snippets returned by search engines. The clusters are labeled such that the user can identify the topics related to the query. So users can quickly focus on the results of a topic in logarithmic time in contrast to the linear time taken in case of flat ranked list. Users can also refine their search by using the topic labels. As an example of this philosophy, Fig. 1 presents the results for the query "java". Even though the conventional search engines are using diversification techniques as a solution to the lexical ambiguity, we cannot find a search result for "island" in the first few search results, for the query "java". Fig. 1 serves as a proof of concept in this example. Further, diversification tries to list top ten results from diverse topics, but does not facilitate exploring the results related to a topic.

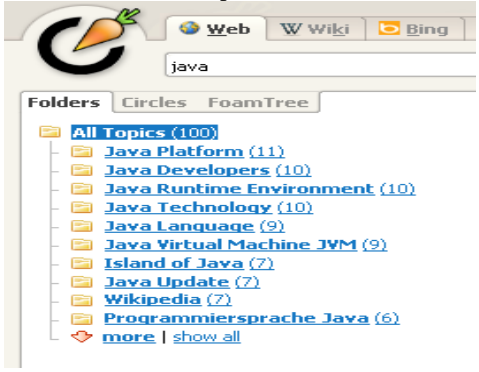

Figure 1. The topics for the query "java" from www.carrot2.org

While SRC is useful for informational [2], polysemous and short queries, it is indeed a challenging task [4]. The topical clusters must be identified on the fly and the cluster labels must be meaningful. Further, this is an ephemeral clustering. Identifying the diverse topics from the short text snippets, variable number of clusters and coverage of all topics related to the query are the additional challenges.

Some of the applications of search results clustering lie in diversification of search results, e-commerce (e.g. ebay), library and bibliographic portals (e.g. DBLP), museum portals, mobile phone browsers (e.g. CREDO), specialized search engines and portals and Semantic Web.

The basic steps to perform the Search Results Clustering are presented in Fig. 2. These steps are briefly explained in the following.

Figure 2.Steps of Search Results Clustering

The search results of a search engine are acquired by using the search engine's API by sending HTTP requests. All major search engines provide APIs with restrictions on the number of queries per day as a free service and as a paid service without such limitations. By sending a RESTful (Representational State Transfer) request to the public search engine APIs, the results are available in either JSON or XML format. Usually the first 100 results are considered.

The title of each search result along with one or two lines summary (called as snippet) forms a search result document. These are usually preprocessed by tokenization, stemming (Porter Stemmer) and stopword removal. If "bag of words paradigm" is used, then the features are the TF-IDF vectors. Other feature extraction methods are frequent phrases and noun phrases.

Wide ranges of clustering algorithms are applicable to generate clusters. These are classified based on the priority given for cluster construction and label generation [4]. In addition to satisfying the "maximum intra cluster similarity

CONFERENCE PAPER
Two day National Conference on Advanced Trends and Challenges in Computer Science and Applications
Organized by: Shree Vishnu Engineering College for Women, Bhimavaram A.P.
Schedule: 18-19 March 2014

and minimum inter cluster similarity" property, the label for each cluster is important for user exploration. The labels must be human readable and meaningful (The classical clustering algorithms represent each cluster by its centroid, which is usually a real number). As the clustering has to happen online, it must operate in real time. So the SRC algorithm usability is subjected to effective label generation, good response time and coverage of the results.

The clusters can be presented to the user as hierarchical folders, as folder tree (Vivismo [1] and Carrot [2]) and with zooming approach (Grokker [3]). Users prefer textual representation with folder tree to the zooming approach [38]. According to the empirical analysis by [36], the clustering interface offers opportunities for diversified searching.

Cluster validation measures play a vital role, as clustering algorithms discover clusters, which are not known a priori. The evaluation of the SRC approaches is significant due to the strict constraints on its response time and their usability by humans. It is not possible to test the cluster performance and tune its parameters at run time. Hence evaluation of the SRC approach has to happen offline before it is released as web clustering engine. This paper surveys the validation approaches suitable for the Search Result Clustering.

The rest of the paper is organized as follows. In the next section, we present various Search Result Clustering approaches and few successful clustering search engines available online. In section 3, we present the clustering validity indices, techniques available in the literature and their applicability to the SRC. Few techniques which are specific to SRC evaluation are presented. The datasets constructed for SRC evaluation, and the performance of popular algorithms on these datasets is presented in section 4. We conclude in Section 5 by summarizing and providing the trends in Search Results Clustering.

II. SEARCH RESULT CLUSTERING ALGORITHMS

Search Result Clustering requires the cluster labels to be meaningful in addition to construction of clusters. Traditional clustering algorithms do not satisfy this criterion. A wide variety of SRC algorithms are available in the literature and they are classified based on their capabilities and importance given to labeling the clusters. [4] provides a detailed survey of the SRC algorithms, and classified the SRC algorithms as *data-centric, description-aware* and *description-centric*. A brief outline of the SRC algorithms according to this classification is presented here. Finally the recent algorithms based on usage of external resources like Wordnet and Wikipedia are discussed.

Data centric algorithms follow the well-known and proven techniques for clustering numeric data, but their keyword-based centriod representation used for labeling is not sensible for humans. Scatter/Gather [10] is a seminal work in this category and uses agglomerative hierarchical clustering. A version of transactional k-means used by a system called WebCAT [18], clustering with committees [35], agglomerative hierarchical clustering with improved feature selection [28] used in Lassi, Tolerance Rough Set Clustering (TRSC) [32], Divide-and-Merge employed by EigenCluster [8], graph of clusters employed by WhatOnWeb [13] and Association rule centric Clustering of web search results [24] is a non exhaustive list of the algorithms in this category. These methods typically work with the "bag of words" for the search results and the keywords as features. The labels generated by these methods were set of keywords (Scatter/Gather), or one or two keywords. Some of these algorithms used N-grams [32] and substring methods [24] to arrive at meaningful labels.

Description-aware algorithms carefully select the features (do not follow "bag of words" notation) so that they result in meaningful cluster labels. These algorithms use frequent phrases appearing in the search results as the features. Suffix Tree Clustering (STC) used in Grouper [48,49], variants of STC - Hierarchical Suffix Tree Clustering(HSTC) [30], STC with Ngram [45], Extended STC [9], TermRank [17], Findex [23], STC+ and NM-STC [25] and SnakeT [16] come under this category.

Earlier approaches perform clustering first and then generate the labels. The third category approaches are specifically meant for SRC, and they work with *description comes first* strategy. Vivismo, a commercial clustering search engine, introduced this approach. Lingo [33, 34] used by Carrot, SHOC used by WICE system [50], CIIRarchies [27], description centric k-means [46], SRC [51], Discover [26], Learn from Web Search Logs to Organize Search Results [45], Concept lattice based - CREDO system [5] and [20], Automatic extraction of useful facet hierarchies from text databases [11] and Deep Classifier [47] fall under this approach.

Recently, algorithms using external resources like ODP taxonomy, Wikipedia, Wordnet etc are developed. Improving Web Search Result Categorization using Knowledge from Web Taxonomy [22], Inducing Word Senses to Improve Web Search Result Clustering [31], Optimal Meta Search Results Clustering [6], Topical Clustering of Search Results [41], Clustering Web Search Results with Maximum Spanning Trees [14] are some of the recent approaches.

The following are the typical characteristics of the SRC algorithms:

1. Use of "bag of words" or N-grams for representation of search results.
2. The Features can be keywords, frequent phrases, noun phrases or gapped sentences.
3. Formation of clusters is considered as most important and labeling phase depends on the clusters found, or labeling is considered as most important and it guides cluster formation.
4. Length of labels can be one or two keywords or use variable length phrases.
5. Use ontological and lexical resources as external resource or no resource usage and work only with the search result snippets.
6. Clusters hierarchy can be multi level or single level.

III. VALIDATION MEASURES

In contrast to traditional clustering techniques, SRC validation requires verification of improvement in the retrieval performance. Hence, it requires clustering evaluation, subtopic retrieval, cluster label quality evaluation, usability tests, coverage analysis and subtopic reach time methods for validation.

[1] www.vivismo.com

[2] www.carrot2.org

[3] www.grokker.org

CONFERENCE PAPER
Two day National Conference on Advanced Trends and Challenges in Computer Science and Applications
Organized by: Shree Vishnu Engineering College for Women, Bhimavaram A.P.
Schedule: 18-19 March 2014

A. *Cluster Evaluation - Internal Measures*

Internal measures of cluster validity evaluate the clustering algorithm based on the information present in the data set itself. These evaluate how well the clusters fit the data without reference to external information, hence unsupervised. These measure the structural properties of clustering like cluster homogeneity (or *cluster cohesion* – how close the objects in a cluster are), separation from other clusters (how much distinct is a cluster from other clusters) [42].

1) Silhouette Coefficient

Silhouette coefficient [42] combines both cohesion and separation. The silhouette coefficient s_i for an individual object *i* is defined as:

$$s_i = \frac{b_i - a_i}{\max(a_i - b_i)}$$

where a_i is is the average distance of object *i* from all other objects in its cluster, and b_i is the minimum average distance to objects in another cluster. The value of the Silhouette coefficient can vary between −1 and 1. Algorithms that produce its value near to 1 (corresponds to a_i near to 0) are desirable. Negative values for s_i are undesirable (corresponds to $a_i > b_i$).

2) Dunn Index

Dunn index [15] attempts to identify compact and well-separated clusters. It is defined as the ratio between the minimal inter-cluster distance to maximal intra-cluster distance. The Dunn index for *n* clusters is defined as:

$$D = \min_{1\le i\le n}\left\{\min_{1\le j\le n, j\ne i}\left\{\frac{d(i,j)}{\max_{1\le k\le n} d'(k)}\right\}\right\}$$

where $d(i,j)$ represents the distance between clusters *i* and *j*, and $d'(k)$ is the intra cluster distance of cluster *k*. Any type of distance measures can be used for $d(i,j)$ and $d'(k)$, for example, distance between the centroids of cluster *i* and *j*, and diameter of cluster *k* respectively. Algorithms that produce clusters with high Dunn index are more desirable.

3) Davies-Bouldin Index

Davies-Bouldin index [12] is defined as a function of the ratio of the within cluster scatter, to the between cluster separation, a lower value will mean that the clustering is better. Davies-Bouldin index for *n* clusters is:

$$DB = \frac{1}{n}\sum_{i=1}^{n}\max_{1\le j\le n, j\ne i}\left(\frac{S_i + S_j}{d(c_i, c_j)}\right)$$

where S_i and S_j correspond to within cluster scatter for clusters c_i and c_j. The separation between the clusters is (c_i, c_j). This measure is also used to determine the number of clusters.

4) Cophenetic Correlation Coefficient

Cophenetic Correlation Coefficient [42] is a measure of how faithfully a dendrogram preserves the pairwise distances between the original unmodeled data points. With $d(i,j)$ as the distance between objects *i* and *j*, T_i and T_j as the corresponding dendogram model points, and with *d* as the average of the $d(i,j)$, and with *t* as the average of the *t(i, j)*. (*t(i, j)* = the dendrogrammatic distance between the model points T_i and T_j), the Cophenetic Correlation Coefficient *c* is given by:

$$c = \frac{\sum_{i<j}(d(i,j) - d)(t(i,j) - t)}{\sqrt{\left[\sum_{i<j}(d(i,j) - d)\right]^2\left[\sum_{i<j}(t(i,j) - t)\right]^2}}$$

A value of the index close to 0 is an indication of a significant similarity.

B. *Cluster Evaluation - External Measures*

The external measures are used to measure the extent to which a clustering algorithm matches a pre specified external structure (ground truth or gold standard). These measures are supervised in nature.

1) Classification Oriented measures

These are used to measure the degree to which predicted cluster labels correspond to actual class labels [42]. The following subsections use the notations: m_i is the number of objects of cluster *i* and m_{ij} is the number of objects of class *j* in cluster *i*. L is the number of classes. *K* is the number of clusters and *m* is the total number of data points.

a. Entropy

Entropy measures the degree to which each cluster consists of objects of a single class. For each cluster *i*, its class distribution is calculated as the probability that cluster *i* belongs to class *j* as $p_{ij} = m_{ij}/m_i$. The entropy of each cluster is computed as $e_i = -\sum_{j=1}^{L} p_{ij}\log_2 p_{ij}$. The total entropy for the set of clusters, *e*, is the weighted sum of each cluster entropy: $e = \sum_{i=1}^{K}\frac{m_i}{m}e_i$.

b. Purity

Purity is a measure of the extent to which a cluster contains objects of a single class. Using the same notations of previous subsection, the purity of a cluster *i* is $p_i = \max_j p_{ij}$, the overall purity of the clustering is $purity = \sum_{i=1}^{K}\frac{m_i}{m}p_i$.

c. Precision

It is the fraction of a cluster that consists of objects of a specified class. The precision of cluster *i* with respect to class *j* is $P_i = p_{ij}$. The total precision of the clustering is $P = \frac{\sum_{i=1}^{K} P_i m_i}{\sum_{i=1}^{K} m_i}$

d. Recall

It is the extent to which a cluster contains all objects of a specified class. The recall of cluster *i* with respect to class *j* is, $R_j = m_{ij}/m_j$. The total recall of the clustering is $R = \frac{\sum_{j=1}^{L} R_j m_j}{\sum_{j=1}^{L} m_j}$

e. F-Measure

F-Measure F_β [38] is a combination of precision *P* and recall *R*; it measures the extent to which a cluster contains only objects of a particular class and all objects of that class: $F_\beta = \frac{(\beta^2+1)PR}{\beta^2 P + R}$

The parameter β is the weighting factor for the importance of the recall (or precision). In SRC domain, we give more weight to recall (β must not be zero, recall weight increases as β increases).

2) Similarity Oriented measures

These approaches measure the extent to which two objects that are in the same class are in the same cluster and vice versa [42]. The following subsections use the notations: TP is number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives.

a. Rand Index and Adjusted Rand Index

Rand index [37] is a measure of the percentage of correct decisions made by the algorithm. It is computed by the formula:

$$RI = \frac{TP + TN}{TP + FP + FN + TN}$$

False positives and false negatives are equally weighted in Rand Index. This may not be suitable for some clustering

CONFERENCE PAPER
Two day National Conference on Advanced Trends and Challenges in Computer Science and Applications
Organized by: Shree Vishnu Engineering College for Women, Bhimavaram A.P.
Schedule: 18-19 March 2014

applications. The value of Rand Index is dominated by true negatives in case of SRC.

The expected value of the Rand Index of two random partitions does not take a constant value (e.g. zero). Hence, the Adjusted Rand index (ARI) is proposed by [21], which corrects the Rand Index for chance agreement. It assumes the generalized hypergeometric distribution as the model of randomness, i.e., the partitions are picked at random such that the number of objects in the classes (gold standard or ground truth) and clusters are fixed. ARI makes RI to vary according to expectation:

$$ARI(C,G) = \frac{RI(C,G) - E\big(RI(C,G)\big)}{\max RI(C,G) - E\big(RI(C,G)\big)}$$

Where $E\big(RI(C,G)\big)$ is the expected values of the RI; C = (C_1,…, C_m) is the clustering and G = (G_1,…, G_g) is the ground truth. ARI is measured by using contingency table for C and G.

RI ranges between 0 and 1, while ARI ranges between -1 and +1 and is 0 when the index equals its expected value.

Even though ARI assumes a fixed number of objects in each cluster, when the clustering algorithm leads to deterministic clustering, the number of clusters will be same with fixed number of objects. Then ARI is applicable. When the clustering algorithm results in different clusters for different executions on the same dataset, then ARI may not be applicable.

b. Jaccard Coefficient

The Jaccard Coefficient is used to quantify the similarity between the ground truth and the clusters. It takes a value between 0 and 1, 1 means that the two dataset are identical, and 0 indicates that the datasets have no common elements. It is defined by the following formula:

$$J = \frac{TP}{FP + FN + TP}$$

c. Folkes and Mallows index

It computes the similarity between the clusters returned by the clustering algorithm and the ground truth classifications. The higher the value of the Fowlkes-Mallows index, the more similar the clusters and the ground truth classifications are. It can be computed using the following formula:

$$FM = \sqrt{\frac{TP}{TP + FP} \times \frac{TP}{TP + FN}}$$

d. Hubert's Γ statistic

This measure considers the comparison of two matrices: *ideal cluster similarity matrix* and *ideal class similarity matrix*. For objects *i* and *j*, *ij*th entry in these matrices is 1 if both objects belong to same cluster or class respectively, otherwise 0. The correlation of these matrices measures the cluster validity. With X(i, j) and Y (i, j) as the (i, j)th element of the matrices X, Y respectively that we have to compare and N as the number of objects, Γ is computed as follows:

$$\Gamma = \frac{1}{TP + FP + FN + TN} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} X(i,j)Y(i,j)$$

High values of this index indicate a strong similarity between X and Y. Similarly Normalized Γ statistic [19] can also be computed and it takes values between -1 and 1.

e. Cluster Validity for Hierarchical Clustering

This approach is to evaluate hierarchical clustering in terms of a (flat) set of class labels, rather than preexisting hierarchical structure (which often does not exist) [42]. So evaluate that for each class, at least one cluster exists that is relatively pure and includes most of the objects of that class. First, F-measure is computed for each cluster in the hierarchy. Later, for each class, maximum F-measure attained for any cluster is taken. By computing the weighted average of all per-class F-measures, the overall F-measure is calculated. Formally, the hierarchical F-measure is defined as:

$$F = \sum_j \frac{m_j}{m} \max_i F(i,j)$$

where the maximum is taken over all clusters *i* at all levels, m_j is the number of objects in class *j*, and *m* is the total number of objects.

C. *Comparison against Ranked List*

As SRC is proposed to overcome the limitations of plain search engines, we need to evaluate whether these improve the retrieval performance over flat ranked lists. The Classification Oriented measures like Precision and Recall can be used after *linearization* of the clustered results. The results of a high density cluster or optimal cluster can be flattened for this purpose [4]. According to [28], a simple interactive method is more effective. The *reach time* and *Subtopic Reach Time* measures (discussed below) assume that a cluster label will allow the user to choose the right cluster. Hence, these measures provide an upper bound on the true retrieval performance.

a. Reach Time

In [34], an analytic method is proposed based on *reach time*: it models the time taken to locate a relevant document in the hierarchy. When *s* is the branching factor, *d* is the number of levels that must be inspected, $p_{i,c}$ is the position of the i^{th} relevant document in the leaf node of the clustering approach, $p_{i,r}$ is the position of the i^{th} relevant document in the ranked list; reach time of i^{th} document is $rt_{clustering} = s.d + p_{i,c}$. The corresponding *reach time* of ranked list is $rt_{rankedlist} = p_{i,r}$. The averaged *reach time*s of the set of relevant documents can be compared.

b. Subtopic Reach Time

This measure is defined as the mean, averaged over the query's subtopics, of the smallest of the reach times associated with each subtopic's relevant results. It is proposed by [3]. For *n* subtopics, the Subtopic Reach Time of rankedlist:

$$SRT_{rankedlist} = \frac{\sum_{i=1}^{n} \min_j P_{i,j}}{n}$$

where $P_{i,j}$ is the position of the j^{th} relevant result of subtopic *i*.

For clustered list, the reach time of a result is given by the position of the cluster in which the result is contained (*c*) plus the position of the result in the cluster (*r*):

$$SRT_{clusterlist} = \frac{\sum_{i=1}^{n} \min_j (c_{i,j} + r_{i,j})}{n}$$

c. Usage of Search Engine Logs

By comparing the search engine logs to clustering engine logs, [49] proposed that we can compare these approaches and avoid the requirement of a test collection with specified relevance judgments. Few metrics for this comparison are – the number of results followed, the time spent and the click distance. But interpretation of user logs is difficult as it involves multiple users and different search tasks.

d. Usability tests/ User studies

Conducting user studies is a viable alternative to automated evaluation of SRC methods. These are subjective tests while the earlier are objective measures. The user (i.e. subject) performs some kind of information seeking task with the systems being compared, the user session is recorded,

CONFERENCE PAPER
Two day National Conference on Advanced Trends and Challenges in Computer Science and Applications
Organized by: Shree Vishnu Engineering College for Women, Bhimavaram A.P.
Schedule: 18-19 March 2014

and the retrieval performance is typically evaluated measuring the accuracy with which the task has been performed, and its completion time. Such studies are especially useful to evaluate inherently subjective features or to gain insights about the overall utility of the methods being tested [4]. Usually, subjects of intermediate web ability are made to participate in the experiment. [7, 43, 16, 23, 3] performed user studies to evaluate improvement in the search experience. [36] studied the search performance and satisfaction level with and without the aid of clusters and hierarchies. This study used a client logging software[4] to record each participant's search process. Mechanical Turk (AMT - Amazon Mechanical Truck)[5] is used in [41] to generate the human ratings. The drawbacks of this methodology are that the tests are not repeatable/replicable, no standards for verification of the user study, dependency on subject's ability and bias. The user studies reported in the literature are favorable to clustering engines.

D. *Quality of Cluster Labels*

Each cluster label indicates the contents of the cluster; hence, meaningful labels are required for user exploration. Salient phrase ranking is proposed in [50], it measures precision of list of labels associated with the clusters assuming that relevance of labels has been manually assigned for each topic. Cluster labels can be assessed by verifying the keywords in the cluster labels with the manually assigned topic labels. The quality of a label can be measured by its informativeness [27], by measuring its relationship to cluster content.

SnakeT [16] used Precision at top N, $P@N = \frac{M@N}{N}$, where M@N is the number of labels which have been manually tagged relevant among the N top-level labels. This measure reflects the user behavior for cluster hierarchy navigation. N values beyond 10 are not considered as users do not like to browse a wider cluster hierarchy.

E. *Subtopic Retrieval*

For a given query, we need to assess whether all documents relevant to the subtopics are retrieved (*kSSL*) and the number of subtopics retrieved (S-recall@K).

1) kSSL

To evaluate the retrieval performance of SRC, Subtopic Search Length under *k* document sufficiency (*kSSL*) is proposed in [2]. It is defined for both ranked lists and clustered results, thus facilitates comparison between search engine's result and clustered result. It measures the average number of items (labels or results) that must be examined before finding a sufficient number (*k*) of documents relevant to any of the query's *n* subtopics. If *k* documents could not be found with the clustering approach, then the user switches back to the ranked list. Hence, this measure models the users' search behavior.

For a ranked list, the value of *kSSL* is simply given by the mean of the ranks of the k^{th} results relevant to each subtopic in the ranked list associated with the query:

$$kSSL_{list} = \frac{\sum_{i=1}^{n} p_{i,k}}{n}$$

Where $p_{i,k}$ is the rank of the k^{th} result relevant to subtopic *i*.

For clustered results, *kSSL* definition involves both cluster labels that must be scanned and the snippets that must be read. Clusters whose labels are relevant to the subtopic at hand are considered. This seperates *kSSL* from the other methods which usually assume that the user is able to select the relevant documents irrespective of the cluster labels. With these considerations, the formula is a sum of three terms: the rank of the last cluster, of the *m* clusters with a relevant label that were visited before retrieving *k* results (denoted $c^*_{i,k}$), plus the sum of the cardinalities of the first $(m-1)$visited clusters, plus the rank of the k^{th} relevant result in cluster $c^*_{i,k}$ (denoted $r^*_{i,k}$). Formally:

$$kSSL_{clusters} = \frac{\sum_{i=1}^{n}\left(c^*_{i,k} + \sum_{j=1}^{m-1} |c_{i,j}| + r^*_{i,k}\right)}{n}$$

When the clusters with relevant labels exhaust before finding k relevant documents, the full ranked list of documents have to be considered. So, the number of search resluts that need to be considered has to be added to the above summation.

The minimum value of *kSSL* depends on the number of subtopics. The topics with more subtopics will have a higher minimum value for *kSSL*, and its value increases with *k*. A computaionally intesive procedure to normalize search lengths over the number of subtopics is proposed in [52]. By considering the weighted average of subtopics, popular subtopics can be given more importance.

2) S-recall@K

A measure of diversification, *subtopic recall-at-K (S-recall@K)*, can be used to used to evaluate SRC techniques. S-recall@K is given by the number of different subtopics retrieved for query *q* in top *K* results returned:

$$S-recall@K = \frac{|\bigcup_{i=1}^{K} subtopics(r_i)|}{K}$$

where $subtopics(r_i)$ is the set of subtopics manually assigned to the search result r_i and M is the number of subtopics for query *q*. This measure is suitable for systems returning ranked lists, so the clusters have to be flattened to a list as given by [14, 31].

F. *Coverage Analysis*

This measures the number of results that are clustered by the SRC method. The results, which do not come under any cluster (orphans), are kept in a separate cluster "other". The size of this special cluster must be as small as possible.

G. *Assessing the significance of Cluster Validity Measures*

The cluster validity measures must be interpreted in statistical terms to avoid the possibility that the observed value is achieved by random chance [42]. The measured value must be statistically significant, as we are interested in clusters that reflect non-random structure of the structure in the data. This analysis can be performed by using Monte Carlo method to compute the probability distribution function of the validity measures [53].

IV. TEST COLLECTIONS

All the supervised validation measures require test data sets with the ground truth for validation. Three test data sets are available for this purpose - AMBIENT[6], MORESQUE[7] and ODP-2398, all of these datasets are freely downloadable.

AMBIENT (AMBIgous ENTries) is a dataset designed for evaluating subtopic information retrieval. It consists of 44

[4] http://www.techsmith.com/morae.asp

[5] crowdflower.com interface to AMT

[6] http://credo.fub.it/ambient

[7] http://lcl.uniroma1.it/moresque/

[8] http://credo.fub.it/odp239

CONFERENCE PAPER
Two day National Conference on Advanced Trends and Challenges in Computer Science and Applications
Organized by: Shree Vishnu Engineering College for Women, Bhimavaram A.P.
Schedule: 18-19 March 2014

topics, each with a set of subtopics and a list of 100 ranked documents. The topics were selected from the list of ambiguous Wikipedia entries. The 100 documents associated with each topic were collected from a Web search engine as of January 2008, and they were subsequently annotated with subtopic relevance judgments.

Table I presents the results of *kSSL* for few popular and efficient SRC methods. *kSSL* for the flat ranked list is presented as baseline (the default results of a search engine Yahoo!), to show the effectiveness of the SRC approaches. The results are taken from TOPICAL [41] (which is most recent SRC method), which compared its results against Lingo [33], Lingo3G (it is a commerical improvement over Lingo and Carrot[2] search uses it), OPTIMSRC [6]. Low values for *kSSL* are desirable, as this measure reflects the time taken by the users to satisfy their information need.

Table I. Evaluation of SRC systems over AMBIENT dataset using kSSL measure

System	*1SSL*	*2SSL*	*3SSL*	*4SSL*
Baseline	22.47	34.66	41.96	47.55
Lingo	24.40	30.64	36.57	40.69
Lingo 3G	24.00	32.37	39.55	42.97
OPTIMSRC	20.56	28.93	34.05	38.94
TOPICAL	17.10	24.02	27.41	30.79

MORESQUE (MORE Sense-tagged QUEry results), is another dataset of 114 ambiguous queries which is developed as a complement to AMBIENT. It is created with an aim to study the behavior of web search algorithms on queries of different lengths, ranging from 1 to 4 words. MORESQUE provides dozens of queries of length 2, 3 and 4 while the AMBIENT dataset is composed mostly of single-word queries. It is created with the 100 top results from Yahoo!, and each query results are annotated as in the AMBIENT dataset. The average Rand Index values as given in [14], are shown in Table II.

Table II. Evaluation of SRC systems over AMBIENT and MORESQUE using average RandIndex

System	AMBIENT	MORESQUE
STC	61.48	51.52
Lingo	62.75	52.68
KeySRC	66.49	55.82
MST	81.53	86.67

The S-recall@K (with K = 3, 5, 10, 15, 20) calculated on AMBIENT + MORESQUE is reported in Table 3. MST [13] performs best, with a subtopic recall greater than all other systems. We observe that KeySRC performs worse than Yahoo! with low values of K, and better with higher values of K.

ODP-239 is another dataset designed for evaluating subtopic information retrieval. It consists of 239 topics, each with a set of about 10 subtopics and a set of about 100 documents associated with single subtopics. The topics, subtopics, and their associated documents were selected from the Open Directory Project [9]. The highest F_1 measure reported so far is 0.413 by TOPICAL [41]. Even though this value is low, the classification on this data set is a hard problem [6]. ODP-239 contains very short documents and the subtopics are very similar to each other. In Table III, we show the F_1 measure of the same SRC methods as given in Table I, except Baseline (it is impossible to find F_1 measure of the search engine flat ranked results).

In Table IV, the mean Silhouette coefficient values obtained using the aforementioned test collections are shown.

Table III. Evaluation of SRC systems on ODP-239 using F1 measure

System	F1 measure
Lingo	0.273
Lingo 3G	0.311
OPTIMSRC	0.313
TOPICAL	0.413

Table IV. Evaluation of SRC systems on Ambient and ODP-239 using Silhouette Coefficient

Data set	Lingo	Lingo3G	KeySRC	OPTIMSRC
AMBIENT	0.22	0.14	0.21	0.27
ODP-239	0.19	0.15	0.20	0.25

Other than these two test collections, evaluations based on user studies used their own test data sets, built by using few popular queries. These studies usually gathered the first 100 results from public search engines like Google and Yahoo!

V. CONCLUSION

Organizing search results into clusters/categories allows users to focus on items in categories of interest rather than having to browse through all the results sequentially. In this paper we have presented Search Result Clustering algorithms available in the literature and the evaluation measures from the perspective of SRC methods. Earlier SRC algorithms were light weight as they work with search results themselves. The recent approaches for SRC use external resources and generate high quality cluster labels.

We have presented the evaluation measures to assess the cluster homogeneity, verification against the ground truth and the measures which are specific to SRC – comparison against ranked list, label quality, user studies etc. We conclude that more user studies are required to uncover the situations where SRC is appropriate.

VI. REFERENCES

[1] Bernardini, A., Carpineto, C., and D'Amico.: M., Full-Subtopic Retrieval with Keyphrase-Based Search Results Clustering. In Proceedings of Web Intelligence 2009, Milan, Italy, pages 206–213. IEEE Computer Society, 2009.

[2] Broder, A.: A taxonomy of web search. In SIGIR Forum 36, 2002

[3] Carpineto, C., Mizzaro, S., Romano, G., and Snidero, M. 2009. Mobile information retrieval with search results clustering: Prototypes and evaluations. J. Amer. Soc. Inform. Sci. Tec. 60, 5, 877–895.

[4] Carpineto, C., Osinski, S., Romano, G.,Weiss, D.: A survey of web clustering engines. ACM Computing Surveys 41(3), pages: 1–38 (2009)

[5] Carpineto, C. and Romano, G. 2004. Exploiting the potential of concept lattices for information retrieval with CREDO. J. Univ. Comput. Sci. 10, 8, 985–1013.

[6] Carpineto, C., Romano, G., Optimal meta search results clustering, SIGIR '10 Proceedings of the 33rd international

[9] www.dmoz.org

ACM SIGIR conference on Research and development in information retrieval, Pages 170-177, ACM New York, NY, USA ©2010.

[7] Chen, H. and Dumais, S. 2000. Bringing order to the Web: Automatically categorizing search results. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM Press, 145–152.

[8] Cheng, D., Vempala, S., Kannan, R., and Wang, G. 2005.: A divide-and-merge methodology for clustering. In Proceedings of the 24th ACM Symposium on Principles of Database Systems, C. Li, Ed. ACM Press, 196–205.

[9] Crabtree, D. , Gao, X., and Andreae, P.: Improving web clustering by cluster selection. In Procs of the IEEE/WIC/ACM Intern. Conf. on Web Intelligence (WI'05), pages 172-178, Compiegne, France, September 2005.

[10] Cutting, D. R., Pedersen, J. O., Karger, D., and Tukey, J.W. 1992. Scatter/Gather: A cluster-based approach to browsing large document collections. In Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM Press, 318–329

[11] Dakka, W. and Ipeirotis, P.G.: Automatic extraction of useful facet hierarchies from text databases. In Procs of the 24th Intern. Conf. on Data Engineering, (ICDE'08), pages 466-475, IEEE Computer Society Washington, DC, USA ©2008.

[12] Davies, D.L. and Bouldin, D.W. : A cluster separation measure. 1979. IEEE Trans. Pattern Anal. Machine Intell. 1 (4). 224-227.

[13] Di Giacomo, E., Didimo, W., Grilli, L., And Liotta, G. 2007. Graph visualization techniques for Web clustering engines. IEEE Trans. Visual. Comput. Graph. 13, 2, 294–304.

[14] Di Marco, A., Navigli, R.,: Clustering Web Search Results with Maximum Spanning Trees, AI*IA'11 Proceedings of the 12th international conference on Artificial intelligence around man and beyond Pages 201-212 Springer-Verlag Berlin, Heidelberg ©2011

[15] Dunn, J.C.: Well separated clusters and optimal fuzzy partitions. *J.Cybern*. 4. 95-104. 1974.

[16] Ferragina, P. and Gulli, A. 2005. A personalized search engine-based on Web-snippet hierarchical clustering. In Proceedings of the 14th International Conference on World Wide Web. ACM Press, 801–810.

[17] Gelgi, F., Davulcu, H. and Vadrevu, S.: Term ranking for clustering web search results. In 10th Intern. Workshop on the Web and Databases, (WebDB'07), Beijing, China, June 2007

[18] Giannotti, F., Nanni, M., Pedreschi, D., and Samaritani, F. 2003. WebCat: Automatic categorization ofWeb search results. In Proceedings of the 11th Italian Symposium on Advanced Database Systems (SEBD), S. Flesca, S. Greco, D. Sacc` a, and E. Zumpano, Eds. Rubettino Editore, 507–518.

[19] Halkidi, M., et al., On Clustering Validation Techniques, Journal of Intelligent Information Systems, 17:2/3, 107–145, 2001 Kluwer Academic Publishers.

[20] Hotho, A., Staab, S., And Stumme, G. 2003. Explaining text clustering results using semantic structures. In Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases. Lecture Notes in Computer Science, vol. 2838. Springer, 217–228.

[21] Hubert, L. and Arabie, P.: Comparing Partitions. Journal of Classification, 2(1):193-218, 1985

[22] Jinarat, S. Haruechaiyasak, C. Rungsawang, A. Improving Web Search Result Categorization using Knowledge from Web Taxonomy, In Proc. of Intern. Conf. on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, ECTI-CON 2009, IEEE, pages 726-730.

[23] Kaki. M., Findex: properties of two web search result categorizing algorithms. In Procs of the IADIS Intern. Conf. on World Wide Web/Internet, Lisbon, Portugal, October 2005.

[24] Kommanti, H.B., Raghavendra Rao, C., Association rule centric Clustering of web search results, MIWAI'11 Proceedings of the 5th international conference on Multi-Disciplinary Trends in Artificial Intelligence Pages 159-168 Springer-Verlag Berlin, Heidelberg ©2011

[25] Kopidaki, S., Papadakos, P., Tzitzikas, Y.: STC+ and NM-STC: Two Novel Online Results Clustering Methods for Web Searching. WISE '09 Proceedings of the 10th International Conference on Web Information Systems Engineering Pages 523 - 537 Springer-Verlag Berlin, Heidelberg ©2009

[26] Kummamuru, K., Lotlikar, R., Roy, S., Singal, K., And Krishnapuram, R. 2004. A hierarchical monothetic document clustering algorithm for summarization and browsing search results. In Proceedings of the 13th International Conference on World Wide Web. ACM Press, 658–665.

[27] Lawrie, D. J., Croft, B. W., and Rosenberg, A. 2001. Finding topic words for hierarchical summarization. In Proceedings of the 24th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM Press, 349–357.

[28] Leuski, A. 2001. Evaluating document clustering for interactive information retrieval. In Proceedingsof the 10th International Conference on Information and Knowledge Management. ACM Press, 33–40.

[29] Maarek, Y. S., Fagin, R., Ben-Shaul, I. Z., and Pelleg, D. 2000.: Ephemeral document clustering for Web applications. Tech. rep. RJ 10186, IBM Research.

[30] Maslowska, I. 2003. Phrase-based hierarchical clustering ofWeb search results. In Proceedings of the 25th European Conference on IR Research, (ECIR). Lecture Notes in Computer Science, vol. 2633. Springer, 555–562.

[31] Navigli, R. Crisafulli, G. Inducing word senses to improve web search result clustering, EMNLP '10 Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing Pages 116-126, Association for Computational Linguistics Stroudsburg, PA, USA ©2010.

[32] Ngo, C. L. and Nguyen, H. S. 2004.: A tolerance rough set approach to clustering Web search results. In Proceedings of the Knowledge Discovery in Databases: PKDD. Lecture Notes in Computer Science, vol. 3202. Springer, 515–517.

[33] Osinski, S., Stefanowski, J., and Weiss, D. 2004. Lingo: Search results clustering algorithm based on singular value decomposition. In Proceedings of the International Intelligent Information Processing and Web Mining Conference. Advances in Soft Computing. Springer, 359–368.

[34] Osinski, S. and Weiss, D. 2005. A concept-driven algorithm for clustering search results. IEEE Intell.Syst. 20, 3, 48–54.

[35] Pantel, P. And Lin, D. 2002.: Document Clustering With Committees. In Proceedings of the 25th ACM International Conference on Research and Development in Information Retrieval. ACM Press,199–206.

[36] Pu, Hsiao-Tieh.,Chen, Sih-Ying.,Kuo, Pei-Yi.: An empirical evaluation on textual results clustering for web search, Proceedings of the American Society for Information Science and Technology, Volume 46, Issue 1, pages 1–16, 2009

[37] Rand, William M.: Objective Criteria for the Evaluation of Clustering Methods. Journal of the American Statistical Association, 66(336):846– 850, 1971.

[38] Rijsbergen, K.V., Information Retrieval,Butterworth-Heinemann, 1979.

[39] Rivadeneira, W., Bederson, B.B.: A Study of Search Result Clustering Interfaces: Comparing Textual and Zoomable User Interfaces, Tech. rep. HCIL-TR-2003-36, University of Maryland.

[40] Sanderson, M.: Ambiguous queries: test collections need more sense. In: Proc. of SIGIR 2008, Singapore, pp. 499–506 (2008).

[41] Scaiella, U., Ferragina, P., Marino, A., Ciaramita, M., Topical Clustering of Search Results, WSDM '12 Proceedings of the

fifth ACM international conference on Web search and data mining Pages 223-232, ACM New York, NY, USA ©2012

[42] Tan, P.-N., Steinbach, M., Kumar, V., Introduction to Data Mining, chapter 8: Cluster Analysis: Basic concepts and algorithms, pages 532-554. Pearson Addison Wesley, 2006.

[43] Turetken, O. And Sharda, R. 2005. Clustering-based visual interfaces for presentation of Web search results: An empirical investigation. Inform. Syst. Front. 7, 3, 273–297.

[44] Wang, J., Mo, Y., Huang, B., Wen, J., and He, L., Web Search Results Clustering Based on a Novel Suffix Tree Structure. In Procs of 5th Intern. Conf. on Autonomic and Trusted Computing, (ATC'08), volume 5060, pages 540-554, Oslo, Norway, June 2008.

[45] Wang, X. And Zhai, C. 2007. Learn from Web search logs to organize search results. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM Press, 87–94.

[46] Weiss, D. 2006. Descriptive clustering as a method for exploring text collections. Ph.D. thesis, Poznan University of Technology, Pozna ´ n, Poland.

[47] Xing, D., Xue, G.R., Yang, Q. and Yu, Y.: Deep classifier: Automatically categorizing search results into large-scale hierarchies. In Procs of the Intern. Conf. on Web Search and Web Data Mining, (WSDM'08), pages 139-148, Palo Alto, California, USA, February 2008.

[48] Zamir, O. And Etzioni, O. 1998. Web document clustering: A feasibility demonstration. In Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM Press, 46–54.

[49] Zamir, O. And Etzioni, O. 1999. Grouper: A dynamic clustering interface to Web search results. Comput. Netw. 31, 11–16, 1361–1374.

[50] Zhang,D. Anddong,Y. 2004.: Semantic, hierarchical, online clustering ofWeb search results. In Proceedings of 6th Asia-PacificWeb Conference (APWeb). Lecture Notes in Computer Science, vol. 3007. Springer, 69–78.

[51] Zeng, H.-J., He, Q.-C., Chen, Z., Ma, W.-Y., And Ma, J. 2004. Learning to cluster Web search results. In Proceedings of the 27th ACM International Conference on Research and Development in Information Retrieval. ACM Press, 210–217.

[52] Zhai, C., Cohen, W. W. and Lafferty. J.: Beyond Independent Relevance: Methods and Evaluation Metrics for Subtopic Retrieval.In *Proceedings of the 26th International ACM SIGIR Conference on Research and Development in Information Retrieval,Toronto, Canada*, pages 10–17. ACM Press, 2003.

[53] Zhu, Li-Xing., Nonparametric Monte Carlo Tests and their Applications, Lecture Notes in Statistics, Vol.182. Springer.