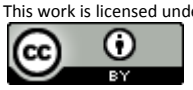
Available online at: https://ijact.in

Date of Submission	03/11/2018
Date of Acceptance	08/12/2018
Date of Publication	28/12/2018
Page numbers	2957-2961 (5 Pages)



ISSN:2320-0790

An International Journal of Advanced Computer Technology

EFFECTIVE PRESENTATION OF RESULTS USING RANKING & CLUSTERING IN META SEARCH ENGINE

Jyoti Mor[1], Naresh Kumar[2], Dinesh Rai[3]

[1]Ph.D. Research Scholar, School of Engineering & Technology, Ansal University, Gurugram, India
[2]Associate Professor, Department of Computer Science & Engineering, MSIT, Janakpuri, New Delhi, India
[3]Associate Professor, School of Engineering & Technology, Ansal University,Gurugram, India

Abstract: The web is changing momentarily which makes it very difficult for the user to retrieve relevant results as per the given query. Clustering is a technique to organize search results in a way so that same search results are associated only with one cluster. For clustering of web pages, different parts of the webpage can be used. There are the lot of algorithms like K-means, Apriori, Expectation maximization, Ada etc. are used for clustering of documents. Clustering Algorithm such as K-means suffers from various problems such as less efficiency and clusters with large entropy. This paper overcomes the problems of K means and makes the use of bisecting K-means algorithm as the primary clustering algorithm having linear time.

Keywords: Meta Search Engine, Search Engine, Webpage, Clustering and Ranking Meta Search Engine.

I. INTRODUCTION

A Meta Search Engine (MSE) takes input from the user and produces results which are gathered from other search engines (SE) [1]. In brief, MSEs are having a single interface with multiple searches [2]. The search results received are an aggregate result of multiple searches. While this strategy gives a broader scope for the search over a single SE, but the results are not always better. Often, the results returned by an MSE are not as relevant since each SE uses its own algorithm to choose the best result [3]. Thus, the idea of clustering the results of a MSE should be introduced to overcome this problem. The commonly used MSE are Dogpile (shows data from Google and Yahoo), Sputtr (get results from many SEs and standalone websites), Clusty (includes various major SEs), and many more. Clustering is an important text mining technique that helps a user to effectively summarize, organize and navigate documents. Document clustering can be used to organize the results returned by an SE in a better way, by organizing a large number of documents into meaningful clusters. The intra-cluster distance must be minimized by having maximum inter-cluster distance. In short, the main benefits of clustering are the arrangement of data into groups having similar characteristics [4] [5].

Clustering has major applications web document classification, image processing, spatial data analysis, market research, pattern recognition, data analysis, biology and many other domains. The K-means is a very popular algorithm for clustering the search results. The algorithm is simple and has low time complexity. The combination of hierarchical and K- means clustering produces Bisecting K-means [3] clustering. A cluster is divided into sub-clusters to obtain the desired number of clusters. After the formation of clusters, it is necessary to rank the documents in each cluster. Clustering without ranking leads to irrelevant or low quality of results. There are many algorithms for ranking such as title snippet, Position merge algorithm etc. Section 1

gives an introduction of MSE and Clustering. Section 2 explains the related work on MSE. Section 3 described the existing problems as identified by related work. Section 4 explains the proposed architecture of Clustering and Ranking MSE and pseudo code. Section 5 describes the experimental results and discussions. Section 6 concludes efficiency of MSE in terms of cluster generation.

II. RELATED WORK

A new method is introduced by the author of [1] to produce clusters of web documents. URL, Title Tag and Meta Tag terms are used for processing. These parts are selected by authors because they contain keywords which are sufficient to describe the whole webpage (WP). K-means clustering algorithm is used for cluster generation. The results obtained by experimental work proved that a cluster formed contains more similar documents than any other cluster. The advantages of this method were that it results into the maximum inter-cluster distance and minimum intra-cluster distance.

Parallel Bisecting K-means including Prediction (PBKP) is proposed in [6], for message-passing multiprocessor systems. Bisecting K-means provide better results when the value of k is extremely large. Every data set and k centroids are used in computation at each iteration.
Whereas, data points of a single cluster and two centroids are used in bisecting K-means, therefore reducing computation time. Similar sized clusters are produced by bisecting k-means while K-means produces clusters of different sizes. Smaller size clusters with smaller entropy are generated by bisecting K-means algorithm. The algorithm PBKP uses the concept of data-parallelism to reduce the workload and improve the speed of the system. At the end, the authors show that the speed of PBKP with data points and with different number of processor is linear in nature
Authors of [3] clustered the web-log data using K-means and Bisecting K-means algorithm, Identical IP address and packet combinations are used for formation of clusters. The clustering framework was further used to detect intrusion from log files. System is tested to check intrusion after the system was first trained by labelling the classes. The security of collected data was maintained by checking whether the given IP addresses are "safe" or "infected". Experimental results show that bisecting k-means performs better than k-means. BKM produced uniform clusters and also empty clusters are not generated by it. Moreover, it took less time for computation; accuracy is more and more efficient when numbers of clusters are increased.
An intelligent cluster search engine (ICSE) is proposed in [7] that uses a fast tree-based search algorithm and Meta-Directory tree as knowledge MSE, meta-directory tree, web pages clustering and topic generation module are main modules of the proposed system.

Following are the characteristics of ICSE:

1) The system combines the results from MSE and information from directories like Google, ODP and Yahoo provide more fruitful results.
2) The system can handle large data sets because it has access to MSE and directory.
3) Computation time is reduced due to the usage of directories.

Retrieve results are ranked in [8] using the combination of Title/Snippet Merge Algorithm (TMA) and Position Merge algorithm (PMA). The PMA and TMA are used to calculate the final score and results are presented to the user in decreasing order of score. The success of the proposed merging algorithm was tested by using TREC-style average precision technique. As a result, authors claimed that merging algorithm can improve the quality of searching.

III. EXISTING PROBLEMS

Based on the above-discussed literature survey some challenges in existing MSEs are as:

1) The results of MSEs are not presented to the user in an effective manner [9].
2) MSEs like yippy [10] produce cluster labels on the basis of the highest frequency of the term that exists within the group but in actual when the clusters are searched according to their labels then found results are different or does not match with the clusters name. Hence result in irrelevant clustering of documents.
3) The relation between the two documents is computed on the basis of terms that exist within these documents but the semantic relation between these documents is not considered which produces irrelevant results [11].
4) Currently available Clustering techniques produce overlapping clusters [12].
5) Now a day's Websites and Ad pays to SE to get a relevant position in search results therefore increasing irrelevant content.
6) In MSE [13] user required to have knowledge about query format for querying multiple SE. But it is very difficult to learn and use a new interface every time.

An MSEs called MEOW proposed in [14] works only on single word query which may result in ambiguous results.

IV. PROPOSED MSE

The proposed architecture for Ranking and clustering the MSE (RnCMSE) is shown in Fig.1. It consists of the following modules:

1) **Downloader:** It will download search results from different MSEs.
2) **Content Extractor:** Contents of a WP is extracted by this module and stored in a text file for further processing. Tokenization is performed here to remove punctuation marks from a document.
3) **Tag Extractor:** Title tag, meta tag and URL tag from each WP are retrieved and stored in a file.

These tags are used because they better define any page as compared to any other data of the WP. There is no need to analyze the whole WP content if these tags are used. Title tag represents the title or purpose of the WP, URL tag represent the web address of the WP and meta tag is part of WP that does not have its existence to end user but contains useful information about it.

4) **Stop Word Remover:** It is used to remove the stop words and punctuation marks from the results of tag extractor.
5) **Stemmer:** In stemming crumple words are reduced to their word stem. Stemmed word and root word may or may not be same. In stemming starting or end portion of a word is removed to form the base word. During stemming some words may be damaged. Porter's algorithm for stemming is used here.
6) **Tf-Idf Calculator:** This module calculates the Tf-Idf value of a WP. As the number of times a term in a document increases its significance also increases. This value helps to determine the relationship between two documents. Tf stands for term frequency which determines how many times a term appear in a document. A WP may contain many irrelevant terms, so to optimize this inverse document frequency is used.

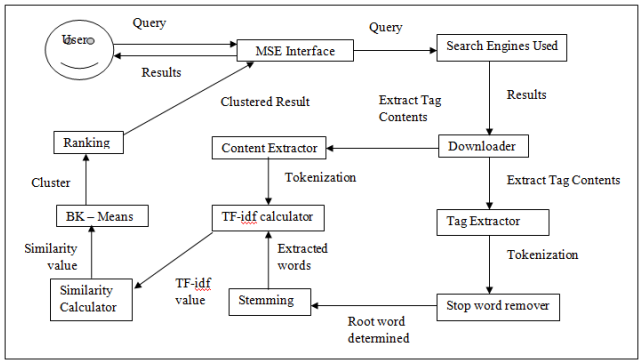


Figure-1: RnC MSE

$Tf = cnt / N$ ... (i)

Where, cnt = term count

N = number of total terms in a document

$Idf = \log(N/Nt)$ ...(ii)

Where, N = number of documents

Nt = Number of documents in which a term appear.

$TfIdf = Tf * Idf$ ... (iii)

7) **Similarity Calculator:** This module determines the degree of match between search results and user query. Tf-Idf values are used to calculate similarity score for each WP.
8) **BK-means:** The base algorithm used for implementation of RnCMSE is Bisecting K-Means algorithm. The main objective of this algorithm is to place all similar objects in a single cluster.

The following are the steps that are followed in Bisecting K-Means algorithm:

i) Select initial cluster for splitting.
ii) Bisecting step is performed in which two sub clusters are formed using basic K-means concept.
iii) Step ii is repeated until highest level of similarity is found between clusters.
iv) Repeat step i, ii and iii until required number of clusters are formed. Further the clusters and results within the clusters are sorted using title/snippet algorithm.

Following are the assumption for the below explained pseudo code.
Input:
N – number of search results
K – number of clusters
Output:
K ranked clusters contains N documents

Step 1: [Downloading of search results]
Download search results from each search engine and remove duplicate results.
Step 2:[Extraction of webpage content]
For each downloaded result
2.1 Extract Content by using tokenization technique.
End for
Step 3: [Extraction of webpage tags]
For each downloaded result
3.1 Extract meta tag, URL tag and meta tag of each search result and store it in a text file D.
End for
Step 4: [Removing of stop words]
For each term in file D
4.1 Extract stop and duplicate words
End for
Step 5: [Perform stemming]
For each term in file D
5.1 apply stemming by using porter's algorithm
End for
Step 6: [Calculation of Tf-Idf]
For all $terms_i$ in D
For all search results j
6.1 $Tf\text{-}Idf_{ij} = Tf_{ij} * Idf_i$
End for
End for
Step 7: [Calculation of simialrity score]
For each search result j
For all terms in D
7.1 $Tf\text{-}Idf_j = Tf\text{-}Idf_j + Tf\text{-}Idf_{ij}$
End for
End for
Step 8: [Apply Bisecting K-Means]
8.1 Place N documents into a single cluster called C
8.2 For each i=1 to K-1 do
For each j=1 to ITER do
Apply K-means

8.2.1 divide C into C1 and C2 sub-clusters
8.2.1 If (intra-cluster similarity score (C1) > intra-cluster similarity score (C2))
C=C2
8.2.3 Else
C=C1
End If
End for
End for
Step 9: [Perform Ranking]
For each cluster C
9.1Sort C clusters using similarity score
End for

V. EXPERIMENTAL RESULTS & DISCUSSIONS

Proposed MSE is tested on different queries of different domain which are medical, technical, general, cars and law. Six queries from each domain is selected to test the proposed work. It is found that clusters produced are more similar in terms of documents that exist within them. Each cluster contains almost same number of documents therefore maintaining the size of clusters.

Table-1 shows result of query "Alzheimer". First column of the table shows URL of the query & second column specified the relevancy score calculated. Table-2 shows the comparison of RnCMSE with existing MSEs in terms of number of meta search engine used, clustering technique used or not, Similarity of results and features of MSE.

Fig-2 shows the formation of clusters in user interface. Different clusters are represented using different colours. Each point of different cluster is associated with their respective URLs which can be identified using Similarity score.

Table-1 URL and similarity score for query "Alzheimer"

URL	Similarity
https://www.alz.org/alzheimers_disease_what_is_alzheimers.asp	0.0381
https://www.alz.org/10-signs-symptoms-alzheimers-dementia.asp	0.221
https://en.wikipedia.org/wiki/Alzheimer%2527s_disease	0.0468
https://www.alzheimers.org.uk/info/20007/types_of_dementia/2/alzheimers_disease	0.0733
https://www.webmd.com/alzheimers/guide/alzheimers-disease-stages	0.0593
https://www.medicinenet.com/alzheimers_disease_causes_stages_and_symptoms/article.htm	0.0907
http://www.nia.nih.gov/health/alzheimers/basics	0.0788
https://www.mayoclinic.org/diseases-conditions/alzheimers-disease/symptoms-causes/syc-20350447	0.0375
https://medlineplus.gov/alzheimersdisease.html	0.0788
https://www.nhs.uk/conditions/alzheimers-disease/	0.0593
http://www.alzheimer.ca/en/Home/About-dementia/Alzheimer-s-disease	0.053
https://www.alzheimers.net/	0.036
https://www.medicalnewstoday.com/articles/159442.php	0.0468
https://www.alzheimersresearchuk.org/	0.0553

Table.-2 Comparison of RnC MSE with existing MSE

Reference	Available Meta Search Engines	Major SEs	Clustering used or not	Relevant results	Merit(s)
[12]	MetaCrawler	Google,Yahoo,Bing, and others Ask.com, About.com, MIVA	Not	Moderate	Used for image, audio, video, business, personal, news and telephone directory,
[13]	WebCrawler	Uses WWW	Not	Moderate	Text search
[14]	ixQu-ick	In total 14	Not	High	Searching for different purposes
[15]	Apo-calx	Not clear	Not	Low	Not known
[16]	Qksearch	NA	Used	Not known	split search and Blend search
[17]	Open-Text	Google, Yahoo, Bing, Ask, Wikipedia and Open Directory	used	Not known	Artificial Intelligence
[18]	iBoog-ie	Majorly MSN	used	Not known	Customizable search tabs
Proposed	RnCMSE	Google, bing	used	High	Non-overlapping clustering and similar size clusters

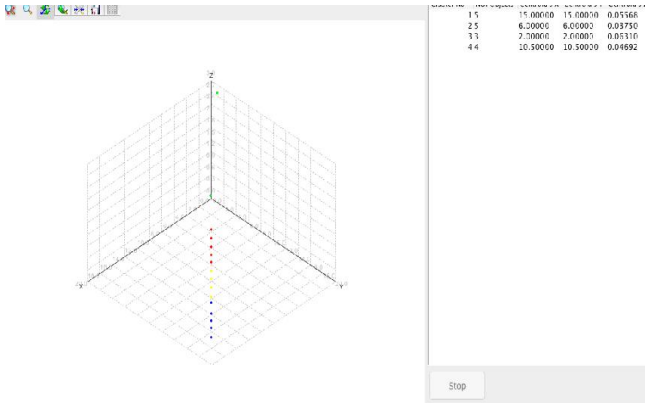


Figure-2: Formation of clusters

VI. CONCLUSION

The RnC MSE uses Google and bing for indexing the results. Java jdk 1.8 environment is used to implement the RnC MSE. MSE is tested for several queries taken from different domain. Internal processing of MSE is easy because it only uses tag content rather than whole WP. The problem with MSE is that it collects results from google and bing therefore it may have low coverage area, this problem will be solved further. Since RnC MSE produces non-overlapping clusters and also removes duplicate links from search results therefore it is efficient in terms of cluster generation. Also ranking is performed using Similarity score therefore user will get relevant clusters at the top.

VII. REFERENCES

[1] S. Poomagal, and Dr. T. Hamsapriya, "K-means for Search Results clustering using URL and Tag contents," International Conference on Process Automation, Control and Computing, pp. 1-7, ISSN: 978-1-61284-764-1, 2011, DOI: 10.1109/PACC.2011.5978906.

[2] Naresh Kumar and Praveer Singh, " Meta Search Engine with Semantic Analysis and Query Processing", International Journal of Computational Intelligence Research, Vol. 13, pp. 2005-2013, ISSN: 0973-1873.

[3] Patil, Ruchika, and Amreen Khan, "Bisecting K-means for Clustering Web Log data," International Journal of Computer Applications 116.19, 2015. URL: https://pdfs.semanticscholar.org/d170/eda4468a4e78ea4ecad0d00620b7a9d135fd.pdf.

[4] Bourair alattar and norita md. Norwawi, "A Personalized Search Engine Based on Correlation Clustering method", Journal of Theoretical and Applied Information Technology, Vol. 93, pp. 345-352, ISSN: 1992-8645.

[5] Naresh Kumar, "Document Clustering Approach for Meta Search Engine", IOP Conf. Series: Materials Science and Engineering, doi:10.1088/1757-899X/225/1/012291.

[6] Li Yanjun, and Soon M. Chung, "Parallel bisecting K-means with prediction clustering algorithm," pp. 19-37, DOI: 10.1007/s11227-006-0002-7.

[7] Chun-Wei Tsai, Ko-Wei Huang, Ming-Chao Chiang, and Chu-Sing Yang, "A Fast Tree-Based Search Algorithm for Cluster Search Engine," Proceedings of the 2009 IEEE international Conference on Systems, Man, and Cybernetics San Antonio, TX, USA, pp. 1603-1608, ISSN: 978-1-4244-2794-9, 2009, DOI: 10.1109/ICSMC.

[8] Yuan Fu-yong, and Wang Jin-dong, "An Implemented Rank Merging Algorithm for Meta Search Engine," International Conference on Research Challenges in Computer Science, 2009, DOI: 10.1109/ICRCCS.2009.56.

[9] N. Kumar, and R. Nath, "A Meta Search Engine Approach for Organizing Web Search Results using Ranking and Clustering," International Journal of Computer, Vol. 10, issue 1, pp.1-7, ISSN: 2307-4531, 2013.

[10] The yippy website [Online] http://www.yippy.com/.

[11] S.Poomagal and Dr. T. Hamsapriya, "K-means for Search Results clustering using URL and Tag contents," International Conference on Process Automation, Control and Computing, pp.1-7, ISSN: 978-1-61284-764-1, 2011.

[12] R. Campos, G. Dias and C. Nunes, "WISE: Hierarchical Soft Clustering of Web Page Search Results based on Web Content Mining Techniques," Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence, pp.301-304, ISSN: 0-7695-2747-7, 2006, DOI: 10.1109/WI.2006.201.

[13] I. Anagnostopoulos, I. Psoroulas, V. Loumos and E. Kayafas, "Implementing a customised meta-search interface for user query personalisation," DOI: 10.1109/ITI.2002.1024655.

[14] Kim Soon Gan, Kim On Chin, Patricia Anthony andVooi Keong Boo, "DBPedia Based Meta Search Engine", Transactions on Science and Technology Vol. 4, pp. 232-251, ISSN 2289-8786, 2017.