International Journal of Advanced Computer Technology (IJACT)

ISSN:2319-7900

CLUSTERING OF WEB QUERY RESULTS USING ENHANCED K-MEANS ALGORITHM

M.Manikantan,
Assistant Professor (Senior Grade), Department of MCA,
Kumaraguru College of Technology,
Coimbatore, Tamilnadu.

Dr.S.Duraisamy
Professor and Head, Dept of MCA,
Sri Krishna College of Engineering & Technology,
Coimbatore, Tamilnadu.

Abstract : **Clustering of web search results is an attempt to organize the web sites into a number of relevant groups. For this process, only documents that match the query are considered while forming the topical groups. Clustering is preformed after the searching process into the resulting documents of the user query. Consequently, the set of related categories is not fixed and they are created dynamically depending on the type of the documents found in the web search results. Also the clustering interface is part of a search engines and it must be done in online. In this paper, we present an efficient clustering algorithm approach and it is enhanced from the k-mean algorithm. In this approach algebraic transformations of the term-document matrix and frequent phrase extraction using suffix arrays are used for clustering the results. Our enhanced K-means algorithm minimizes the processing time and maximizes the cluster count in web search.**

***Keywords – Clustering Algorithm, Web Search, K-Mean Algorithm, Suffix Tree Clustering Algorithm, Processing Time*.**

Introduction

Web search engines have been mostly used to find information in the Web, in which the search results are usually returned as a list of Web pages. However, as usually large numbers of Web pages are returned by a user query, it is very difficult for the users to find the appropriate Web pages in the list. Although there have been a lot of ranking algorithms proposed to improve the searching effectiveness, it increasing the resultant Web data volume in huge. Based on the attributes of the documents, that is processed and grouped by the various clustering algorithms. Clustering of web search results was first introduced in the Scatter-Gather system. During the clustering process, the resultant clusters based on the contents and the links of the document. Aiming at solving this problem, researchers proposed to cluster the search results, in which the search results are clustered in terms of several topics, with each topic contains some related Web pages. Traditional topic clustering approaches only consider the textual relevance between query terms. To partially overcome this problem, query expansion and query refinement techniques are commonly applied with WorldNet. For this purpose, we use more comprehensive resources, the encyclopedia Wikipedia, as the basis for query refinement. Capturing user query context is severely slowed down by the fact that user preference varies in time. With an enormous growth of the Internet it has become very difficult for the users to find relevant documents. This algorithm is easy to implement, requiring a simple data structure to keep some information in each iteration and it is used in the next iteration[1]. Our experimental results demonstrated that our scheme can improve the computational speed of the *k*-means algorithm by the magnitude in the total number of distance calculations and the overall time of computation. Moreover, the internal relationships among the documents are in the search results that are rarely presented and are left for the user. One of the alternative approaches is to automatically grouping the search results into related groups. In response to the user's query, currently available search engines return a ranked list of documents along with their partial content called snippets. If the query is general, it is extremely difficult to identify the specific document which the user is interested .Hence the users are forced to shift through a long list of off-topic documents [2].

Relevance Feedback

The most natural way of obtaining user's subjective information and preferences is by using models that incorporate online learning from the user interactions with the search engine. The basic idea for this model is to integrate the relevance feedback loop into the interaction between the system and the user. The concept of relevance feedback is based on the analysis of the user deciding decisions and preferences [3]. K-means is an iterative algorithm in which clusters are built around K central points are called centroids. The algorithm starts with a random set of centroids and assigns each object to its closest centroid. Then, repeatedly, for each group, based on its members, a new central point is calculated and objects assignments to their closest centroids. The objects are usually described by sets of numerical attributes K is a parameter of the algorithm and must be known before the clustering starts. The algorithm finishes when no object reassignments are needed or when certain amount of time elapses [4].

Input: D= {d1, d2….dn} \\set of n items
Output: A set of k-clusters.
Steps:

1. Arbitrarily choose k-data items from D as initial centroids;
2. Repeat assigns each item di to the cluster which has the closest centroid, calculate new mean for each cluster; until convergence criteria are met.

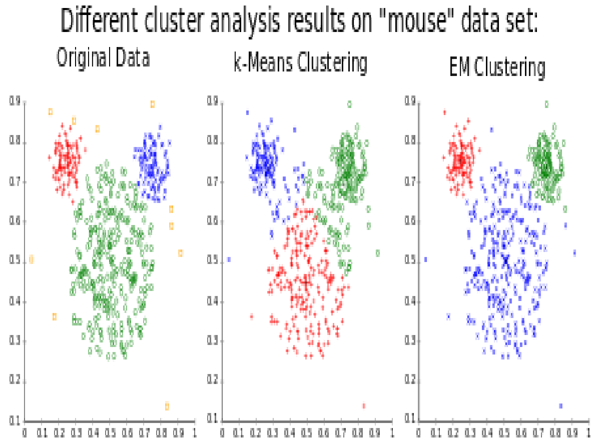

Figure 1. Various Cluster Results

In this example (Fig.1), the result of k-means clustering contradicts the obvious cluster structure of the data set. The small circles are the data points, the four ray stars are the centroids. The tendency of *k*-means to produce equi-sized clusters leads to bad results; the number of clusters in *k* is an invalid output parameter. An inappropriate choice of data sets for *k* may yield poor results. Hence for clustering the online dataset, enhanced k-means Algorithm [5] is more suitable than k-mean algorithm. Here the choice of the optimal value for the parameter keywords ultimately depends on the users' preferences [6].

Clustering Method

A. Preprocessing

The aim of the preprocessing phase is to prune from the input of all characters and terms that can possibly affect the quality of group descriptions. Text filtering removes html tags, entities, non-letter characters except for sentence boundaries. Each snippet language is identified and finally processes the appropriate stemming and stops words removal. The preprocessing phase is automatically generated and summarized of the original documents and hence it is usually very small in one or two sentences.

B. Frequent phrase extraction

This paper uses the SVD-decomposed term document matrix to identify abstract concepts and single subjects or groups of related subjects that are collectively different from other abstract concepts. To be a candidate for a cluster label, a frequent phrase or a single term must be:

- Appear in the input documents at least certain number of times (term frequency threshold)
- Not cross sentence boundaries
- Be a complete phrase
- Not begin or end with a stop word.

C. Cluster label induction

Once frequent phrases and single frequent terms that exceed the term frequency thresholds then they are used for cluster label induction. There are three steps to this: term-document matrix building, abstract concept discovery, phrase matching and label pruning. The term-document matrix is constructed out of single terms that exceed a predefined term frequency threshold. Weight of each term is calculated using the standard term frequency. Singular Value Decomposition method is applied to the term-document matrix to find its orthogonal basis. As discussed earlier, vectors of this basis SVD's matrix represent the abstract concepts appearing in the input documents in the phrase matching and label pruning step, where group descriptions are discovered, relies on an important observation that both abstract concepts and frequent phrases are expressed in the same vector space and the column space of the original term-document matrix A. Thus, the classic cosine distance can be used to calculate how a nearest phrase or a single term is to an abstract concept. Let us denote by a matrix of size t×(p+t) where t is the number of frequent terms and p is the number of frequent phrases. It can be easily built by treating phrases and keywords as pseudo-documents and using one of the term weighting schemes [7].

D. Cluster content discovery

In the cluster content discovery phase, the classic vector space model (VSM) is used to assign the input documents to the cluster labels induced in the previous phase. In a way, we re-query the input document set with all induced cluster labels. The assignment process resembles document retrieval based on the VSM model. Let us define matrix Q, in which each cluster label is represented as a column vector. Let C = QTA, where A is the original term-document matrix for input documents. This way, element cij of the C matrix indicates the strength of membership of the j-th document to the i-th cluster. A document is added to a cluster if cij exceeds the Snippet Assignment Threshold, yet another control parameter of the algorithm. Documents not assigned to any cluster end up in an artificial cluster called other clusters.

E. Final cluster formation

Finally, clusters are sorted for display based on their score, calculated using the following simple formula: C score = label score × kCk, where kCk is the number of documents assigned to cluster C. The scoring function, although simple, prefers well-described and relatively large groups over smaller, possibly noisy ones. For the time being, no cluster merging strategy or hierarchy induction is proposed for this Enhanced k-Mean Algorithm [8].

Proposed Algorithm

A. Preprocessing

```
1: D   input documents (or snippets)
{STEP 1: Preprocessing}
2: for all d 2 D do
3: perform text segmentation of d; {Detect word boundaries etc.}
4: if language of d recognized then
5: apply stemming and mark stop-words in d;
6: end if
7: end for
```

Description: Here we get the input documents as snippets and detect the word boundaries by applying stemming and mark stop-words.

B. Frequent Phrase Extraction

```
8: concatenate all documents;
9: Pc discovers complete phrases;
10: Pf   p: {p 2 Pc ^ frequency (p) > Term Frequency Threshold};
```

Description: Here we concatenate all documents and discover complete phrases by obtaining frequency terms.

C. Cluster Label Induction

```
11: A   term-document matrix of terms not marked as stop-words and with frequency higher than the Term Frequency Threshold;
12: U, V   SVD (A);
{Product of SVD decomposition of A}
```

Description: The term document matrix is marked with stop-words and higher frequencies are composed to a single value. The terms are phrased as matrix and clusters are labeled by enhance k-mean. The cosine similarities between the cluster labels also calculated. Then the one with highest score among similar label is chosen.

D. Enhanced k-mean

```
1: k   0; {Start with zero clusters}
2: n   rank (A);
3: repeat k+1;
4: k   k + 1;
5: q (Pk i=1 _ii)/(Pn i=1 _ii);
6: until q < Candidate Label Threshold;
7: P   phrase matrix for Pf;
8: for all columns of UT k P do
9: find the largest component mi in the column;
10: add the corresponding phrase to the Cluster Label Candidates set;
11: label Score   mi;
12: end for
```

Description: Calculate cosine similarities between all pairs of candidate labels; Identify groups of labels that exceed the Label Similarity Threshold; for all groups of similar labels do select one label with the highest score; k as the minimum value that satisfies the following condition: kAkkF /kAkF _ q, where kXkF symbol. It denotes the frobenius norm of matrix X. Clearly, the larger the value of q the more cluster candidates will be induced. The choice of the optimal value for this parameter ultimately depends on the users' preferences.

E. Cluster Content Discovery

```
13: for all L 2 Cluster Label Candidates do
14: create cluster C described with L;
15: add to C all documents whose similarity to C exceeds the Snippet Assignment Document;
16: end for
17: put all unassigned documents in the "Others" group;
```

Description: Documents are assigned to column matrix by vector space model; the document is assigned with the cluster and the remaining is put in others group; At last snippet assigned documents are retrieved for input.

F. Final Cluster Formation

```
18: for all clusters do
19: clusterScore←labelScore × IICII;
20: end for
```

Description: The clusters are displayed with high score by calculating cluster score formulae.

Results & Discussion

Academic domain was taken into the implementation process and Weka tool was used for the algorithm implementation and clustering the web results. K-mean algorithm and our proposed enhanced K-mean algorithm were implemented into the Weka tool for clustering the given query and the results were compared. Based on the results the enhanced K-means Algorithm is efficiently forms the thematic groups of cluster in minimum response time compare to exiting method. Accordingly one of the example query result is categorized in Table 1.

User Query: "Best Universities in India"

Table1. Cluster result for the user query "Best universities in India"

Cluster No.	Cluster Name	No. of web sites under the cluster
1	"Deemed universities"	32
2	"Government universities"	27

Cluster No.	Cluster Name	No. of web sites under the cluster
3	"Medical universities"	56
4	"Agricultural universities"	23
5	"Foreign universities"	15
6	" Research Universities"	35

Various types of user queries are processed into the academic domain and categorized the final results in Table 2. The query processing time and the number of clustering groups are the two main factors to evaluate the efficiency of the two methods. Based on this result, the enhanced k-means method is more effective and efficient for online data clustering techniques.

Table2. Comparison results in processing time and cluster count

User Query in Academic Domain	No. of Web Sites Used	No. of Clusters		Processing Time	
		Modified K-means	K-means	Modified K-means	K-means
Best Polytechnics	100 sites	7	6	45 Seconds	48 Seconds
Best R&D Research centers	200 sites	9	7	46 Sec	50 Sec
Best schools	300 sites	6	5	48 Sec	53 Sec
Best Universities	400 sites	9	8	50 Sec	58 Sec
Best Engineering Colleges	500 sites	10	9	54 Sec	64 Sec
Best B-Schools	600 Sites	5	4	56 Sec	70 Sec

The graph in Fig.2 shows the processing time for clustering the web sites based on the user query of both methods. The y axis shows the time taken for query processing and clustering the web sites and x axis shows the number of web sites participated in the query process.

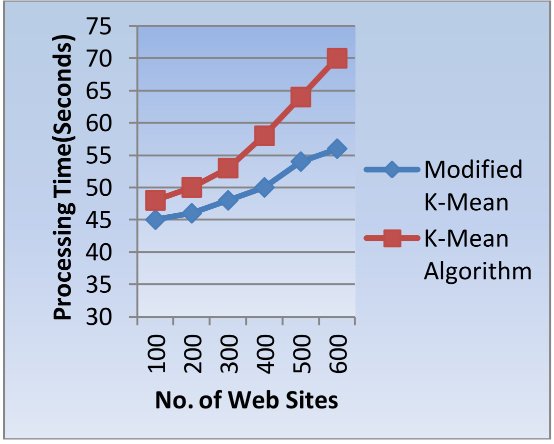

Figure 2. Processing time comparison for modified K-Mean vs. K-Mean algorithm

The graph in Fig.3 shows the cluster count based on the user query of both methods. The y axis shows the number of clusters after the query processed and x axis shows the number of web sites participated in the query process.

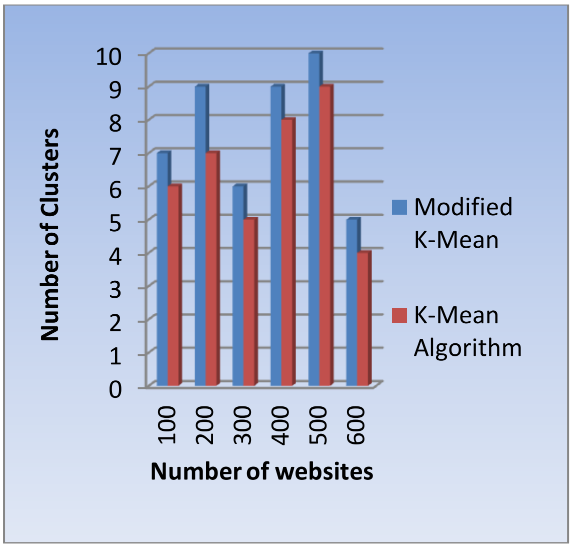

Figure 3. Cluster count comparison for modified K-Mean vs. K-Mean algorithm

Conclusion

One of the most popular clustering algorithms is k-means, but in this method the quality of the final clusters relies heavily on the initial centroids, which are selected randomly. Also the k-means algorithm is computationally very expensive in query cost. The enhanced method also chooses the initial centroids based upon the random selection, but it is very sensitive to the initial starting points and it produce the unique clustering results. In this less similarity based clustering method the initial cluster centers will not be selected randomly, so the accuracy of the result will be high. The experimental results show that proposed algorithm provides the better results for various datasets. In our method the suffix sorting process reduces the running time of our algorithm. Especially in case of web snippets, our method is very effective for thematic filtering.

References

[1] Taher Niknam, Elahe Taherian Fard, Narges Pourjafarian, Alireza Rousta, "An efficient hybrid algorithm based on modified imperialist competitive algorithm and K-means for data clustering," Elsevier, Engineering Applications of Artificial Intelligence, Volume24, Issue2,pp. 306–317, March 2011.

[2] Shi Yu, Tranchevent, Xinhai Liu, Glanzel, Suykens, De Moor and Moreau, Y, "Optimized Data Fusion for Kernel k-Means Clustering," Pattern Analysis and Machine Intelligence, IEEE Transactions on Volume 34, Issue 5, pp.1031-1039, 2011.

[3] Shraddha Shukla and Naganna S, "A Review on K-means data Clustering approach," International Journal of Information & Computation Technology, Volume4, pp.1847-1860, 2014.

[4] K. A. Abdul Nazeer, M. P. Sebastian," Improving the Accuracy and Efficiency of the k-means Clustering Algorithm," Proceedings of the World Congress on Engineering, Vol I, 2009.

[5] Dawid Weiss and Jerzy Stefanowski, "Web search results clustering in Polish: Experimental evaluation of Carrot," In Proceedings of the New Trends in Intelligent Information Processing and Web Mining Conference, 2003.

[6] K. Chandramouli, T. Kliegr, J. Nemrava, V. Svatek and E. Izquierdo, "Query Refinement and User Relevance Feedback for Contextualized Image Retrieval," 5th International Conference on Visual Information Engineering, 2008.

[7] Hua He, Jimmy Lin and Adam Lopez, "Massively Parallel Suffix Array Queries and On-Demand Phrase Extraction for Statistical Machine Translation Using GPUs," Proceedings of NAACL-HLT 2013, pages 325–334, 2013.

[8] Ji-Rong Wen, Jian-Yun Nie & Hong-Jiang Zhang, "Clustering user queries of a search Engine," ACM, pp.162–168, 2001.

Biographies

M.MANIKANTAN is Assistant Professor (Senior Grade) in Department of Computer Applications, Kumaraguru College of technology, Coimbatore. He obtained MCA Degree in Bharathiar University and M.Phil.(CS), in Manonmaniam Sundaranar University, Tamilnadu. He is life member of ISTE (Indian Society for Technical Education) and active member in CSI (Computer Society of India) Coimbatore chapter. His research interests include Web Mining and Database Technologies.

Dr.S.DURAISAMY is Professor and Head of the Department of Computer Applications, Sri Krishna College of Engineering and Technology, Coimbatore. He obtained his PhD degree from Alagappa University, Tamilnadu. He is the author of over 10 journal papers, as well as numerous proceeding papers and technical reports. His research interests are object oriented software metrics and software quality analysis.