Asian Journal of Computer Science and Technology
ISSN: 2249-0701 Vol.7 No.3, 2018, pp. 82-85

Construction of Lexicons to Perk Up Re-Clustering

A. George Louis Raja[1], F. Sagayaraj Francis[2] and P. Sugumar[3]
[1]Research Scholar, Department of Computer Science and Applications,
SCSVMV University, Kanchipuram, Tamil Nadu, India
[2]Professor, Department of Computer Science and Engineering, Pondicherry Engineering College, Puducherry, India
[3]Assistant Professor, Department of Computer Applications,
Sacred Heart College (Autonomous), Tirupattur, Tamil Nadu, India
E-Mail: george@shctpt.edu, fsfrancis@pec.edu, sugumar86@gmail.com

***Abstract* - The existing semantic methods cluster the documents based on unabridged or abridged term comparisons. After clustering, these terms are not preserved, costing the cluster operation to be repeated in its entirety upon the arrival of new documents. Hence the semantic clustering methods can be considered as "*on the go*" methods. Re-clustering becomes unavoidable in all circumstances both in the Iterative and Incremental Clustering Methods. It would be more appropriate to build and evolve a *lexicon* with the derived keywords of the documents and to refer them in further cluster operations. The rationale is to deny re-clustering upon new documents and refer the Lexicon to formulate clusters until the quality of clusters is intact, and when it breaks above the threshold, the cluster operation can be repeated. Since re-clustering is delayed until a breakeven point, the process of re-clustering becomes faster. This process may incur additional runtime complexity, but would extremely simplify and speed up the process of re-clustering. This paper discusses about the construction of lexicons and its applications in clustering. The Keyword based Lexicon Construction Algorithm (KBLCA) is demonstrated to build lexicons and the breakeven point for re-clustering is proposed and described. The theory of denying re-clustering is briefed, along with experimental results.**
***Keywords:* Lexicon, Clustering, ATSCA, Keygraph, KBLCA**

I. INTRODUCTION

The lexicon is considered to be thesauri of words used in a language or domain. Lexicons can be thought to be dynamically growing online dictionaries. They provide indexed access to each word of the lexicons along with the usage of the word. The construction, organization and retrieval mechanisms are the parameters that distinguish the lexicons by measuring the user accessibility. The earlier adaption of lexicons was primarily focused on the natural language processing, artificial intelligent and domain specific applications. The recent developments in semantic clustering has emerged a strategy referred as Gloss based similarity measure augmented by lexicons.

II. KEYWORD BASED LEXICON CONSTRUCTION ALGORITHM (KBLCA)

The major objective of this paper is to present the method for constructing the lexicons obtained as by-products of key word based clustering mechanism. The process can be considered as an extension from clustering to Lexiconization. One of the driving factors to evolve the lexicons is to simplify the process of re-clustering. Keeping together all the above objectives intact, the KBLCA Algorithm is proposed as shown in Figure 1.

A. Phase I: Keyword Extraction

The documents of the input text corpus are put through the *KeyGraph* algorithm to reveal the pertinent keywords in them. The extracted keywords say $k_1,k_2,\dots k_n$ from the document d_i are preserved in a array L_i. This process is iteratively repeated for the entire documents d_i in the text corpus.

B. Phase II: Clustering

Every Keyterm K_i present in the document D_A is mapped with the Key term Kj of the document D_B if $K_i=k_j$, i.e when the terms are distinct. This is done through the ATSCA algorithm which tries to identify the degree of association among the document A to document B, and clusters the documents in the corpus yielding clusters $C_1,\dots C_n$.

C. Phase III: Lexicon Construction

It can be noted that the documents in the clusters are linked through their key terms, hence the set of keywords of each cluster now describe the Lexicons L_1, L_2, ...L_n. The constructed lexicons can be applied in the re-clustering process described in the following section. The stages of Lexicon construction can be better understood by the algorithm shown.

D. Algorithm KBLCA

Input: Dataset A containing the documents
Output: Lexicons $\{l_1,l_2...,l_n\}$from the documents of the dataset A
1. *begin*
2. *for each* document d_a in dataset A
3. *extract* the keywords from the document$\{k_1,k_2,\dots,k_n\}$;
4. *end for;*
5. *for each* document d_a in the dataset A
6. *perform ATSCA_Clustering*in the dataset A;

7. *end for;*
8. *for each* cluster c_i in the dataset *A*
9. *Add* the top-n key terms of d_i in the cluster c_k the lexicon l_j;
10. *end for;*
11. *end;*

III. EXPERIMENTAL RESULTS

The experimental corpus with 300 text documents mentioned in Table I was executed with the KBLCA Algorithm implemented in *R* Language.

TABLE I LEXICONS GENERATED BY KBLCA

Clusters			
Internet of Things	**Text Mining**	**Big Data**	**Software Engineering**
α=134.13, n=18	**α=5622.47, n=44**	**α=655.24, n=42**	**α=351.72, n=37**
Lexicons			
Data	Mine	Mine	System
IOT	Data	Data	Design
Compute	Text	Text	Engineering
Cloud	Information	Information	Development
Architecture	Database	Database	Compute
Connect	Opinion	Opinion	Data
Control	Construct	Construct	Test
Manage	Natural	Natural	Process
Access	Process	Process	Requirement
Design	Require	Require	Product
Application	Department	Department	Program
Approach	Public	Public	Structure
Address	Document	Document	Quality
Collect	Determine	Determine	Network
Case	Number	Number	Active
Machine	Development	Development	Tool
Target	Decision	Decision	Inform
Technique	Database	Database	Control
	Rule	Rule	Technology
	TF	TF	Component
	IDF	IDF	Architecture
	Stem	Stem	Approach
	Warehouse	Warehouse	Communicate
	Item	Item	Interface
	Include	Include	Secure
	Record	Record	Concept
	Represent	Represent	Method
	Discover	Discover	Database
	Review	Review	Profession
	Strategy	Strategy	Algorithm
	Set	Set	Interact
	String	String	Complex
	Support	Support	Effect
	Knowledge	Knowledge	Server
	Follow	Follow	Logic
	Procedure	Procedure	Experience
	Technique	Technique	Degree
	Cover	Cover	Discipline
	Design	Design	
	Methods	Methods	
	Research	Research	

IV. HOLDING-UP RE-CLUSTERING

The premise of the KBLCA algorithm is to prevent re-clustering upon the arrival of a new document in the text corpus. When a new document arrives for clustering, invariably all the text clustering algorithms repeat the clustering process from the scratch, leading to the increased time complexity.

The KBLCA algorithm can avoid re-clustering until the clusters are intact. The stages of this process are depicted in the figure 1. When a new document enters into the text corpus for clustering, the document is clustered based on the reference to the Lexicons $L_1, L_2, ... L_n$. After clustering the cluster cohesion and separation values are compared with a pre-defined threshold.

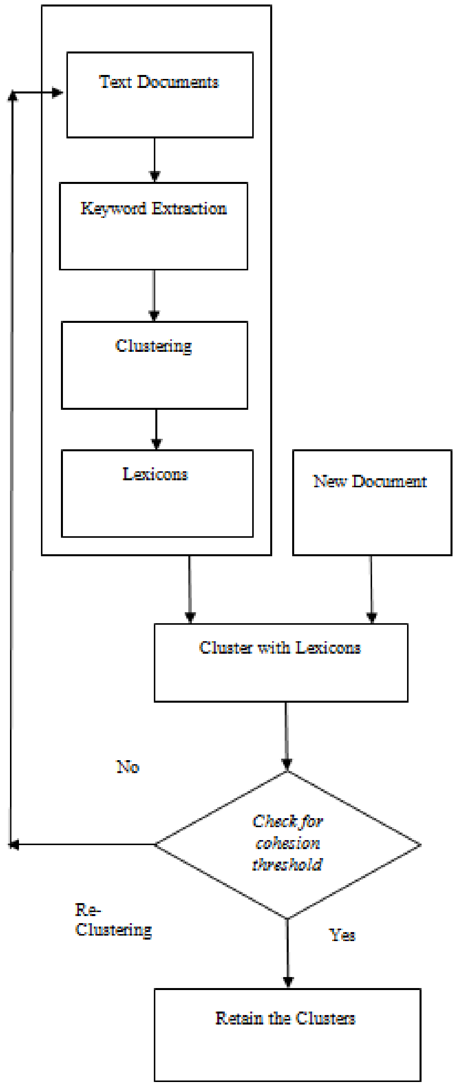


Fig. 1 KBLCA Lexicon Construction Algorithm

To demonstrate how long the process of re-clustering can hold up, the test document corpus containing 300 documents were injected with additional 10 documents in every run of further experiments. Initially the cluster cohesion and separation values were within threshold, and started fluctuating when documents of other categories were given as input breaking above the threshold. On the other hand, when the test documents were in the same category, it was observed that the threshold values were intact.

The Table II presents the cluster cohesion values at every run of the ATSCA algorithm. The observation show that the clusters were intact until the documents were under existing categories, and started disintegrating when documents of new categories were fed as input.

A. Algorithm Deny_Clustering

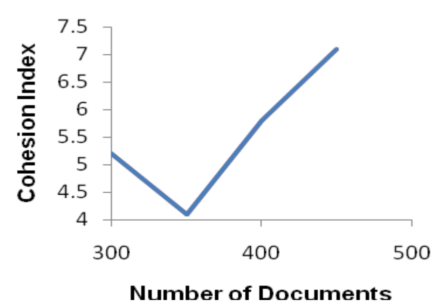
Input: Clustered Dataset *A*with *k* clusters and a new document *d* to cluster
Output: Clusters upon the documents of the dataset *A* and *d*

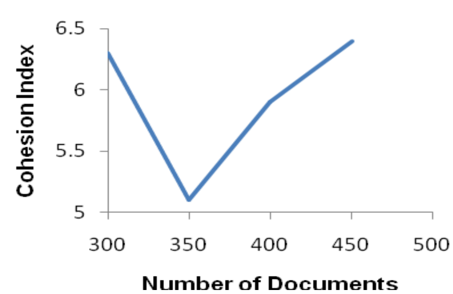
```
begin
    perform tockenizing, stop word removal and
    stemming on d;
    for each Lexiconl in dataset c
        for each term t in document d
            if (L(i) == t) then count(i)++;
        end for;
    end for;
    sort the count(i) values of d;
    assign d in cluster l;
    calculate threshold v= n/2;
    if (k<v) then
            return;
    else
            cal lATSCA_Clustering;
    end if;
end;
```

TABLE II CHANGES IN CLUSTER COHESION

Cluster	Observed Cluster Cohesion			
	Initial	Iteration		
		I	II	III
Internet of Things	5.21	4.12	5.81	7.1
Text Mining	6.31	5.12	5.98	6.42
Big Data	11.34	10.49	12.4	13.2
Software Engineering	9.16	8.14	9.2	10.3

The following Figure 2 represents the deviation of cohesion values in the four clusters generated at the initial phase; at initial iteration and iteration I the values are decreasing, indicating that the cluster quality is improving. But, gradually they started to increase in the iterations II and III, referring to the decline of cluster quality. This experiment indicates that the condition to perform re-clustering is when the cluster cohesion increases constantly.





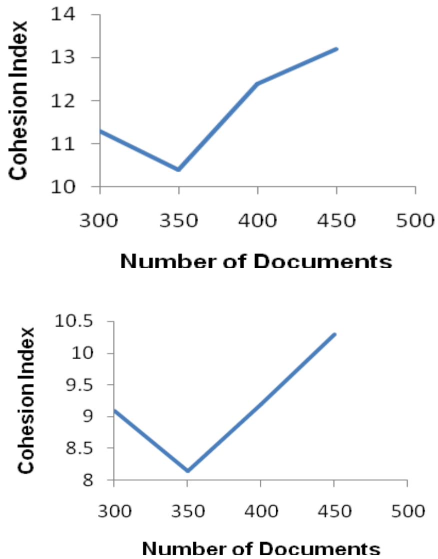


Fig. 2 Deviations in Cluster Cohesion

A similar effect was observed in terms of the cluster separation parameter, the following Figure 5.5 illustrates the observations of separation values. At initial runs the separation was within threshold, and started decreasing when new documents in different categories were introduced. The conclusion is to deny re-clustering until the separation value start to decrease after iterations.

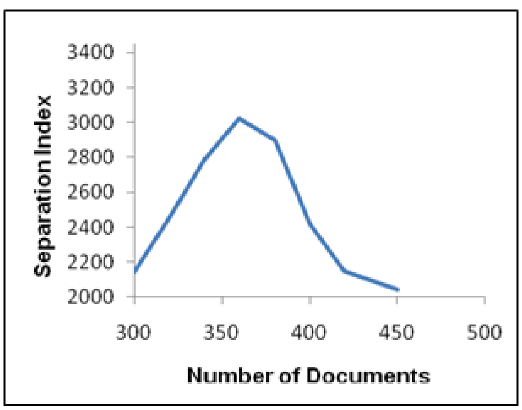


Fig. 3 Deviations in Cluster Separation

Hence, the threshold value for re-clustering is until the cluster quality degrades.

V. CONCLUSION

The study on re-clustering policies explored the prospect to propose a hypothesis to deny re-clustering. This paper proposed a mechanism to delay the re-cluster operation augmented by the Lexicons built from the most conclusive key terms of the clustered documents. Through the proposed breakeven point for re-clustering, it was demonstrated that lexicons are powerful tools to delay re-clustering, and thus speeding and simplifying the re-cluster operation. The scope of clustering is limited to Information Retrieval, but it can also be carefully extended to plagiarism detection and relevance ranking. The Lexicons contributing to the cluster formation can be utilized to derive the partial ordering of documents, which broadens the scope of clustering.

REFERENCES

[1] H. Sayyadi and L. Raschid, "A Graph Analytical Approach for Topic Detection", *ACM Transactions on Internet Technology (TOIT)*, Vol. 13, No. 2, 2013.

[2] Snehalata M. Lad., "Keyword Extraction from Conversation Text Document and Recommending Document using Fuzzy Logic Based Weight Matrix Method", *International Journal of Advanced Research in Computer Science*, Vol. 7, No. 4, pp. 34-38, August 2016.

[3] His-Cheng Chang and Chiun-Chieh Hsu, "Using Topic Keyword Clusters for Automatic Document Clustering", *Proceedings of the Third International Conference on Information Technology and Applications, IEEE*, 2005.

[4] Youngsam Kim, Munhyong Kiml, Andrew Cattle and Julia Otmakhova, "Applying Graph-based Keyword Extraction to Document Retrieval", *International Joint Conference on Natural language Processing*, October 2013, 864-868.

[5] Maryam Habibi and Andrei Popescu-Belis, "Keyword Extraction and Clustering for Document Recommendation in Conversations", *IEEE*, Vol. 23, No. 4, pp. 746-759, 2015.

[6] Mohammad Rezaei, Najlah Gali and Pasi Franti, "CIRank: A Method for Keyword Extraction from web pages using Clustering and distribution of nouns", *IEEE/ WIC /ACM International Conference on Web Intelligence and Intelligent Agent technology*, Vol. 1, pp. 79-84, 2015.