ISSN:0975-9646

Reorganization of Duplicate Data Cleaning and Cluster Generation for Documents

Ajay Kumar[1], Davesh Singh Som[2], Ramander Singh[3]

[1,2,3]*Department of Computer Science and Engineering*, Meerut Institute of Engineering and Technology, *Meerut, Uttar Pradesh, India*

Abstract- This paper proposes a new and efficient methodology for classification of research documents (pdf documents), clean of duplicate data and cluster generation for the documents. The topic the efficient and easiest searching for documents into different clusters makes. This technique can be utilized by search engines to provide relevant results to the user according to query and also utilized by online journal domains that are maintaining large set of documents. This paper also suggests a good cluster generation and word matching technique so, the time consume for finding the appropriate cluster for a document will be reduced. The proper clustering of documents will be further utilized by multi document summarization system, which produces a summary for the documents related to each other.

Keywords: Cluster, clustering, word matching, classification of PDF documents.

I. INTRODUCTION

Concept of Dirty Data: The concept of dirty data can be said as any data which is not consistent with the already residing data in a data warehouse. The types of dirty data could be misspellings like "green" replaced by "rgeen","1" replaced by "l", typographical or phonetic errors. Some fields also have numerical constraints like weight cannot be "negative", people cannot have "more than two parents", a human cannot be of "more than 100-120 years". Some organizations have different formats of field like a date field is of the format DD/MM/YY or MM/DD/YY which leads to a data entry error, while entering the data manually, a different unit of measurement e.g.:" meters" v/s "inches", different modes of payment e.g.: "daily" v/s "weekly" or "monthly" v/s "annually". Another major form of dirty data is duplicity which leads to wastage of resources. Duplicity may be due to spelling variations, naming conventions etc.

Steps involved in data cleansing:-

Data cleansing is usually a two-step process including detection and then correction of errors in a data set.

The steps involved in Data Cleansing are:

1. Identification of errors-records could have incomplete or corrupted data.
2. Perform error verification-whether it is truly an error or not. This situation occurs in organizations where there exists a usage of organizational jargons [2].
3. Extract the data to be cleaned-the data is extracted and stored in a temporary table, operations are performed and the data is repaired and verified, then it is replaced in the target table.
4. Perform data cleaning-which can be done automatically or manually.

Manual process is however avoided as it is highly time consuming and tedious in nature. It is limited by human capabilities like speed, accuracy in error detection and correction. Thus leading to more error prone performances and degrading the quality of data, which in turn leads to increase in operational costs and hence poor decision making.

1. It is extremely important to categorize the data according to the rate of its criticality.
2. Critical errors-needs to be immediately addressed i.e.; error reporting ,verification and cleansing
3. Non-critical errors-can is temporarily ignored.
4. Further section will involve a discussion about the algorithms that can be followed for data cleansing.

Introduction of Cluster Generation:-

Clustering means grouping of documents which are similar to each other into one group. The main uses of clustering of documents are –

1. If a collection is well clustered, we can search only the cluster that will contain relevant Documents.
2. Searching a smaller collection improved effectiveness and efficiency.

Proper clustering of documents in digital library in the form of research papers (pdf files) is required for efficient searching of documents according to the terms. The clustering technique limited the search of the query to a specific set of documents and so the time of the searching to find the relevant document could be saved. This paper discusses the three main issues of clustering –

1. To decide the nature of for comparison with other documents.
2. Selection of algorithm is selected for sending the document to a particular cluster.
3. 3. Increasing the vocabulary dictionary word files, which are used for deciding to which cluster the
4. Document belongs?

This paper suggests creation of cluster keyword file, which contain keywords (or words) related to the documents in the cluster. Cluster keyword file also contains the docID of the documents in which the corresponding keyword is present docID's tells in how many documents the particular keyword is present, which helps further for choosing the appropriate cluster for the document.

The work has been divided into three sections: classification of documents, removal of duplicate documents and automatic generation of clusters. In this documents are sending to the predefined repository by matching the terms of the new document with the terms of the Dictionary word file.

Removal of duplicate documents is checked by first checking the title of the research paper with the title of the papers which are already present in the various clusters. Automatic generation of cluster is achieved by setting the cluster keyword file, which take entries of the different terms (keywords) present in the abstract, introduction and proposed methodology parts of the research document.

II. RELATED WORK

Andrew McCallum and Kamal Nigam presented a paper on "Efficient Clustering of High Dimensional Data Sets with Application to Reference Matching". This paper introduces a technique for clustering that is the client when the problem is large in all of these three ways at once. The key idea is to perform clustering in two stages, first a rough and quick stage that divides the data into overlapping subsets. In the second stage, we execute some traditional clustering algorithm, such as Greedy Agglomerative Clustering or K-means using the accurate distance measure.

Closely related to the above methods are a large number of extensions to and variants on KD-trees such as multi resolution KD-trees [11], which recursively partition the data into subgroups. Almost all of these methods suffer from doing hard partitions, where each item must be on a single side of each partition.

Niall Rooney, David Patterson, Mykola Galushka, Vladimir Dobrynin presented a paper on "A scalable document clustering approach for large document corpora" In this paper, the scalability and quality of the contextual document clustering (CDC) approach is demonstrated for large data-sets using the whole Reuters Corpus Volume 1 (RCV1) collection. CDC is a form of distributional clustering, which automatically discovers contexts of narrow scope within a document corpus.

Giles et al. [12] also study the domain of clustering citations. They present experiments with several different word matching techniques and one string edit based technique. They find that the best-performing method is a word matching algorithm similar to the Cora technique used in Section 3, augmented to use field extraction information and bigrams.

III. PROPOSED METHODOLOGY

I. Classification of documents

Classification of documents means assigning the documents to the predefined repositories as shown in the figure 1.

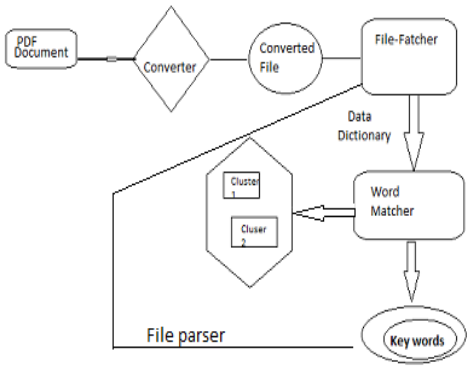


Figure.1 General Architecture of the classification of documents.

It is divided into following sub modules as follows –

1. PdfToText_Convertor:

This model fetches pdf files (research papers) one by one from the repository, convert it into text file and then collect into a Text file repository.

2. File_fetcher:

This module retrieves one by one text file from the text file repository and supplied it to the Starting_Center_Line_Reader module.

3. Word_matcher:

This module separate out the words from the sentences by tokenizing the sentences by considering space and comma as a separation criteria of words , now these words matches with the words of the Dictionary word files The dictionary file is organized into two fields in which one field contain the word belonging to the topic and another contain match hit which counts the number of times the word matches.

II. Duplicate documents Cleaning Approach

Removal of duplicate documents means no two documents of the same contents included in the cluster. Removal of duplicate documents is shown in figure 2 –

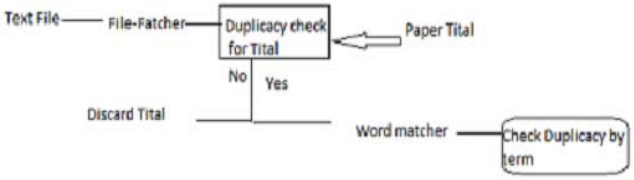


Figure.2 General Architecture for duplicate document removal.

Removal of duplicate documents is done with the help of two sub modules-

1.Check_duplicacy_by_title

First of all, the first word of the title of the new document is matched with the first word of the first entry of the repository "title_of_papers". And if the first word matche, then the second word of the title of the new document matches with the second word of the first entry this process of matching of words of title of new document is repeated until or unless whole of the words of the first entry of the repository not matched with the entire words of the title of the new document. And if mismatch occurs, then the same process of matching words of the title of the new document, matched with the rest of the entries of the repository.

2. Check_duplicacy_by_term

This module ensures that no two documents of the same contents reside in the single repository. The terms of the document first of all matches with the cluster's keyword file, as shown below-

S.No	TERM	DOC_Id
1	Unicast	1,6,9,15,78,84
2	Multicast	4,8,13,18,92,116
3	Broadcast	7,12,16,23,39,83
4	ISDN	22,34,55,76

Table.1 Cluster keyword (term) file.

Suppose the term is available in the cluster keyword file, then the terms of the new document matches with the terms of the documents, whose docid is given corresponding to the term.

III. AUTOMATIC GENERATION OF CLUSTERS

The automatic generation of cluster is the process of maintaining a cluster keyword file, which contains the keywords(terms) related to the documents for a given cluster, along with terms with one more field is maintained in the cluster keyword file that accounts the docid's of the documents in which the term is available.

Procedure of Automatic Document Clustering:

There is not a single operation from the collection of documents to the clustering of document collection. It includes the number of stages that consist generally four main stages:-

1. **Preprocessing:** Before document representation, we require some preprocessing. Firstly, we need to remove stop words such as „a", „any", „the", since they are frequent and irrelevant. Secondly, we need to stem the words. For eg. „Flying" and „flew" are stemmed to „fly".
2. **Feature Extraction:** It employs to produce the set of features by parsing each document. It helps to remove the noise and reduce the dimensionality of feature space. The most commonly used feature selection metric are term frequency and inverse document frequency.
3. **Document Representation:** Most of the clustering approaches use the vector space model for document representation. In VSM, the document is represented as the vector of keywords. A collection of n documents with m unique words is represented as an m*n matrix, where each document is a vector of m dimension.
4. **Document Clustering:** At this stage, the target documents are grouped into different clusters on the basis of selected features.

Pseudo code of the automatic generation of cluster

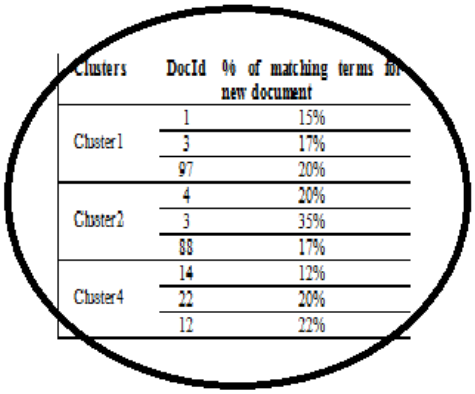
1. **Cluster_list[] contain the list of all clusters, such as cluster1 means cluter[1], cluster2 means cluster[2] and so on.**

2. **Extract terms of the new document excluding terms of references, conclusion, result part of the document into term_list_newdocument[]**

```
For (all elements of cluster_list[])
{
   For (all elements of  term_list_newdocument[])
     {
        doc_list[] ← document which contains term[i]
        for(all elements of doc_list[])
          {
            For(all terms"cluster keyword file" of cluster[i])
                  {
                     If term € doc_list[i]
                     {
                       Term_list[] ← term

                     }
                  }
       If ( more than 20% matches occurs between terms of  term_list[] and term_list_newdocument[] )
              Write cluster no. and % of matching into a file
          }
     }
}
```

The cluster whose document has highest % of matching of terms with the terms of the new document is chosen as a appropriate cluster for the document. If there exists clusters whose documents have equal % of matching of terms with the terms of the new document, then we look the cluster which have more number of documents whose % of matching of terms with the terms of the new document is more than or equal to 20%, then such a cluster is an appropriate cluster for the new document. The table below shows the term(i) of new document doc(i) is available in documents of cluster1, 2 and 4 and its % matching of terms with the documents of the cluster is also given. By seeing this , we make a conclusion that doc(3) of cluster 2 has highest % of matching of terms with the terms of the new document doc(i), so, cluster 2 is the appropriate cluster for doc(i).



Clusters	DocId	% of matching terms for new document
Cluster1	1	15%
	3	17%
	97	20%
Cluster2	4	20%
	3	35%
	88	17%
Cluster4	14	12%
	22	20%
	12	22%

Figure3. % of matching of terms of documents with term of new Documents

IV. CONCLUSION

This paper suggests a idle technique for clustering of pdf documents also suggests creation of dictionary word files, which contain keywords (or words) related to a particular topic for which a cluster is created. The matching of words with the words of the dictionary word file is done by extracting the words of the starting and middle sentences of the file and matched with the words of the dictionary word file. The searching method used by word matcher algorithm makes searching faster as there is no need to search all of the words of the dictionary. The knowledge base of the system is increased by fetching the keywords, which are mentioned in the file under the heading "keywords or key terms" and storing them into dictionary word file.

REFERENCES :-

[1] C. Aggarwal, S. Gates, and P. Yu. On the merits of building categorization systems by supervised clustering. In Proceedings of (KDD) 99, 5th (ACM) International Conference on Knowledge Discovery and Data Mining, pages 352–356, San Diego, US, 1999. ACM Press, New York, US

[2] R. Agrawal, C. Aggarwal, and V. V. V. Prasad. Depth-first generation of large item sets for association rules. Technical Report RC21538, IBM Technical Report, October 1999.

[3] R. Agrawal, C. Aggarwal, and V. V. V. Prasad. A tree projection algorithm for generation of frequent item sets. Journal of Parallel and Distributed Computing, 61(3):350–371, 2001.

[4] Deepti Gupta,Komal Kumar Bhatia, A.K. Sharma, A Novel Indexing Technique for Web Documents using Hierarchical Clustering, IJCSNS International Journal of Computer Science and Network Security,VOL.9 No.9, September 2009.

[5] F. Beil, M. Ester, and X. Xu. Frequent term-based text clustering. In Proc. 8th Int. Conf. on Knowledge Discovery and Data Mining (KDD)'2002,Edmonton,Alberta,Canada,2002.http://www.cs.sfu.ca/~ester/publications.html.

[6] S. Chakrabarti. Data mining for hypertext: A tutorial survey. SIGKDD Explorations: Newsletter of the Special Interest Group (SIG) on Knowledge Discovery & Data Mining, ACM, 1:1–11, 2000.

[7] M. Charikar, C. Chekuri, T. Feder, and R. Motwani. Incremental clustering and dynamic information retrieval. In Proceedings of the 29th Symposium on Theory Of Computing STOC 1997, pages 626–635, 1997.

[8] P. Domingos and G. Hulten. Mining high-speed data streams. In Knowledge Discovery and Data Mining, pages 71–80, 2000.

[9] R. C. Dubes and A. K. Jain. Algorithms for Clustering Data. Prentice Hall College Div, Englewood Cli®s, NJ, March 1998.

[10] S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering data streams. In IEEE Symposium on Foundations of Computer Science, pages 359–366, 2000.

[11] A. Moore. Very fast EM-based mixture model clustering using multi resolution kd-trees. In Advances in Neural Information Processing Systems 11, 1999.

[12] C. L. Giles, K. D. Bollacker, and S. Lawrence. Cite Seer: An automatic citation indexing system. In Digital Libraries 98 – Third ACM Conference on Digital Libraries, 1998.