Automatic Text Categorization of Marathi Documents Using Clustering Technique

Mrs. Sushma R. Vispute*, Prof. M. A. Potey**

*PG Student, DYPCOE, Akurdi, Pune, India, **PG Faculty, DYPCOE, Akurdi, Pune, India

*visputesushma@gmail.com, **mapotey@gmail.com

Abstract— **The purpose of the present work is creating an intelligent system to retrieve desired documents in Marathi language. The system also focuses on providing the personalized documents in Marathi language to the end user based on their interests identified from the browsing history.**

This paper presents the automatic categorization of Marathi documents and the literature survey of the related work done in automatic categorization of text documents. Several supervised learning techniques are exists for the classification of text documents namely Decision trees, Support Vector machine (SVM), Neural Network, Ada Boost and Naïve Bayes etc. Several clustering techniques are also available for text categorization namely K-means, Suffix Tree Clustering (STC), Semantic Online Hierarchical Clustering (SHOC), Label Induction Grouping Algorithm (LINGO) etc. In the literature survey it is found that vector space model (VSM) gives better result than probabilistic model.

This paper presents categorization of the Marathi text documents using Lingo Clustering algorithm based on VSM. The data set consists of 107 Marathi documents of 3 different categories- Tourism, Health Programmes and Maharashtra festivals. The result shows that the performance of the LINGO clustering algorithm is good for categorizing the Marathi text documents. For the Marathi documents overall accuracy of the system is 91.10%.

Keywords: Text categorization, Clustering, Information filtering, Internet search, Information retrieval.

I. INTRODUCTION

The Web based search engine is a popular place for gathering information due to the fast growth of the Internet. There are lot of sources of retrieving information, such as company websites, personal homepages and organization websites, etc. The useful secret information stored in large databases is used to assist the decision making process of an association. However, the number of data administrators and analysts rises at a lesser rate than the amount of stored data. So, there is a need for automatic techniques for extracting useful information from the huge data. And also with the fast use of the Internet it is not easy for the end user to search required documents in preferred language. For the general query, it becomes hard for the user to identify interested document. The users have to go through a huge set of documents.

Text document categorization is one of the important steps in the data mining and information retrieval (IR) field. Effective searching in the large amount of documents available on the Internet is a time consuming process. Categorizing documents manually is also a time consuming process and requires more precision.

For making searching of the document an easy task, there is a need to do document categorization automatically. Several techniques are available for document categorization: Decision trees, Support Vector machine, Rule induction, Neural Network, Ada Boost and Naïve Bayes [11], [12], [13]. Another approach is automatically grouping of documents using clustering techniques. Several clustering techniques are also available for text categorization namely K-means, Suffix Tree Clustering (STC), Semantic Hierarchical Online Clustering (SHOC), LINGO etc. [5], [6], [7], [9].

A. Motivation

As searching the document on the Internet is a time consuming process, this system saves user's effort required to perform document categorization manually and retrieving documents automatically. The present work aims at developing a system for retrieving desired documents in Marathi language. This system helps Marathi people those are from Marathi medium and Marathi background for Marathi content generation and focuses on providing personalized documents in Marathi language to the end user based on their interests identified from the browsing history and hitting count.

B. Overview

In this paper, section 2 presents the related work for the different languages; Section 3 describes the text categorization approaches; Section 4 outlines the proposed work and the system architecture and section 5 describes the experimental setup and results. Section 6 includes conclusions and future enhancements.

II. AN OVERVIEW OF RELATED WORK

This part of the paper discusses the literature survey of related work for document categorization and content retrieval done by the various authors.

Since text document categorization is a main step in the data mining and information retrieval (IR) field, several research papers' authors have focused on automatic categorization of the text documents and automatic content retrieval. There are many previous work invented in this area. The most of the work is done for the automatic categorizing English and Latin text documents.

978-1-4673-2818-0/13/$31.00 ©2013 IEEE

Authors Kohilavani S [1] and E. Iniya Nehru [2] et al. applied Navie Bayes algorithm for automatically classifying Tamil text documents and getting useful information from those documents to create the knowledge base.

Author ElKourdi et al. [3] used *Navie Bayes* algorithm for classifying the Arabic web documents automatically, the results shows that the average accuracy is 68.78%.

Author Saleh Alsaleem [4] applied Support Vector Machine (SVM) and Navie Bayes algorithms for automatically classifying the Arabic text documents. The result for the different Arabic data sets shows that SVM algorithm gives better result than the NB algorithm.

At last Stanislaw Osinski et al. [6], [7], [9] applied Label Induction Grouping (LINGO) clustering algorithm on Web documents. They found 70-80% clusters useful to the user and 80-95% of the snippets into these clusters matching with their query.

III. TEXT CATEGORIZATION APPROACHES

Several supervised and unsupervised learning techniques are exists for the classification of the text documents namely Decision trees, Support Vector Machine, Neural Network, AdaBoost and Naïve Bayes [11], [12], [13]. Several clustering techniques are also available for text categorization namely K-means, Suffix Tree Clustering (STC), Semantic Online Hierarchical Clustering (SHOC), Label Induction Grouping Algorithm (LINGO) etc. [5]. In this section the main focus is on LINGO algorithm.

A. Clustering for Text Documents

Clustering technique forms the groups of similar items from the input documents' set. The feature of high-quality clusters is that items into the same cluster are similar to each other, and these are unrelated for two dissimilar clusters. Let us discuss some approaches to the text clustering.

1) Hierarchic Agglomerative Clustering (HAC):

Every step of the HAC algorithm merges an item/items and a cluster or two clusters' that are similar to each other into a new cluster and the association between items are symbolized in a dendogram which is similar to tree [9].

2) *K-means algorithm*:

In this algorithm the number of input clusters are need to specify. It is an iterative method. Clusters are made in the region of *c* centre points. It begins with the arbitrary set of the centre points and adds the each item to its nearest centre point. Then, iteratively, for the every group, it computes a new centre point and changes item assignments to their nearest centre points if needed. The algorithm ends when no items reassignments require [9].

"Suffix Tree Clustering (STC)", "Semantic Hierarchical Online Clustering (SHOC)" and LINGO are created for the general query. The main focus is on LINGO because of its' lots' of features. Drawbacks of the STC are: 1) does not good for the bigger high class phrases, 2) the less instructive for smaller phrases. So if the document does not contain required phrases, it will not be displayed in the output even though it is useful and related. By supporting two concepts', "complete phrase" which is left and right complete and a "continuous cluster definition", SHOC overcome the drawback of the STC. The drawback of the SHOC is the use of fuzzy threshold value the cluster explanation. Also many times it generates insensitive continuous clusters [9].

The most of algorithms forms the cluster first, and then as per the content of the cluster; it assigns the name (label) to the cluster. LINGO algorithm is having a lot of advantages over the other clustering algorithms such as it supports dynamic clustering as per the user query instead of static one, it identify cluster label first then assigns the document to that cluster, it is based on vector space model, it can be work for multiple languages and supports multiple keyword based searching. LINGO algorithm overcomes the drawback of SHOC algorithm by first creating a user understandable cluster label and then adds documents to that cluster. Lingo first find outs user readable frequent words/phrases from the documents. Then uses Singular Value Decomposition (SVD) method to reduce the term document matrix, and then discovers the titles of clusters and then based on the similarity value assigns documents to that cluster titles.

B. LINGO Clustering Algorithm

Algo_LINGO shows the main phases of Lingo algorithm [7].

Algo_LINGO: Main phases of the Lingo algorithm

```
        Input: $Dc$ = set of documents
        Output: Clusters of documents.
{PHASE 1: Pre processing of documents}
        [Apply pre processing to each document]
        For each $d$ € $Dc$ do
        {
                Apply stop words removal process to $d$;
                Apply stemming process
        }
{PHASE 2: Extraction of "Frequent Phrases"}
    [Find frequent keywords and "frequent phrases"]
    1)  $Cp$ = complete_phrases discovery;
    2)  $Fp$ = $p$: {$p$ € $Cp$ □ frequency (p) >
        Term_Frequency_Threshold};
{PHASE 3: Cluster Title/Label discovery}
    [Use SVD to discover cluster title]
        1)  $M$ = term_document_matrix
        2)  $S, U, V$ = SVDT ($M$);
        3)  $k = 0$; {initialise with zero clusters}
        4)  $n = RANK(M);$
        5)  Repeat
                i)   $k = k + 1;$
                ii)  $p = (\sum_{i=1}^{k} \Sigma ii)/ ( \sum_{i=1}^{n} \Sigma ii);$
            Until $p$ is less than Threshold of Candidate_Label;
        6)  $P = Fp$ phrase_matrix;
        7)  For column space of $U_k^T P$
            {
                i)   Find the largest column element $l_i$ ;
                ii)  Add phrase to the Candidate set of
                     Cluster_Label; $labl\text{-}Scor = l_i;$
            }
        8)  Using cosine ranking space, work out similarities
            between all candidate_label pairs.
        9)  Find out the labels exceeding the Threshold of Label
            Similarity ;
        10) For each set of related labels
            {
```

```
            Choose 1 label having the maximum score;
        }

{PHASE 4: finding of Cluster documents}
   [Use VSM to determine the documents of clusters]
      1)   For each CL € Candidates of Cluster_Label
           {
              i)    Construct cluster Cr relating CL;
              ii)   Assign to Cr all documents whose
                    similarity exceeds the Threshold;
           }
      2)   Add all remaining documents to the separate group
           "Others" ;
{PHASE 5: Discovery of final Cluster}
   [Calculate scores of clusters and perform merging]
      For each cluster calculate
      {
           $Clust\text{-}Scor = labl\text{-}Scor \times \|Cr\|$;
      }
```

In the first step LINGO algorithm apply pre processing techniques to the documents to improve the efficiency of clustering process. It includes removing of stop words those are not useful for clustering and apply stemming techniques for finding the root words;

In the second step LINGO extracts "Frequent phrases" from the documents and these phrases are nothing but the chronic ordered sequences of the terms occurring in the documents. A single keyword or a "frequent phrase" must satisfy the following to become a part of the label of a cluster:

1. It should repeat in the documents at least definite number of times
2. It should be a "complete phrase" that is left and right complete.
3. It should not start with stop word and also not end with a stop word.

In the third step LINGO discovers labels of the clusters and for doing this it uses frequent phrases satisfying thresholds of word frequency. Using *TF-IDF* (Term Frequency- Inverse Document Frequency) weighting function it computes "Term Document" matrix and then by applying Singular Value Decomposition technique to this matrix discovers labels of the clusters.

SVD divides "Term Document" matrix M into 3 matrices V, S and U, as $M = U\ S\ V^T$, *where*, S is a $t \times d$ "diagonal matrix", U is a $t \times t$ "orthogonal matrix" and V is a $d \times d$ "orthogonal matrix" *[9]*. The number of non-zero singular values of matrix M represents its rank r_M. For the column space of M, the first r_M columns of U form an orthogonal basis. The phrase having the maximum value in the vector is selected as the user comprehensible concept. In addition, the cosine ranking value becomes the score of the candidate of the label of a cluster.

In the fourth step LINGO discovers documents of the Clusters. For doing this the cluster labels discovered in the previous step is passed to this phase where, the Vector Space Model is applied to allocate the input documents to the labels of clusters. Let in matrix A, column vector shows the cluster label. Let $E = A^TM$, where M represents the "Term Document" matrix of the given documents. Factor e_{ij} of the E matrix represents the relationship of the j^{th} document with the i^{th} cluster. If e_{ij} exceeds the threshold, a document is added to a cluster. Remaining documents not added to any cluster is added to the separate *cluster "Others" [9].*

In the last step LINGO discovers final clusters having maximum scores. Based on the ranking score of the clusters they are sorted for display and provide to the user, cluster score is calculated using the following formula:

Clust-Scor = *labl-scor* × $\|Cr\|$, where $\|Cr\|$ is the total number of documents of the cluster *Cr*.

IV. OVERALL SYSTEM ARCHITECTURE

Several authors of the research papers have focused on automatic categorization of the text documents and automatic information retrieval for various languages. For example English, Latin, Spanish, Tamil, Polish and Arabic.

The present work expands the work in [1], [2], [3], [6], [7] for Marathi documents using Lingo clustering algorithm based on Singular Value Decomposition and Vector Space Model, which helps the Marathi people for Marathi content generation.

A. System Description:

The detail architecture of the system is as shown in the figure 1, which shows the various blocks of the system and their functionalities.

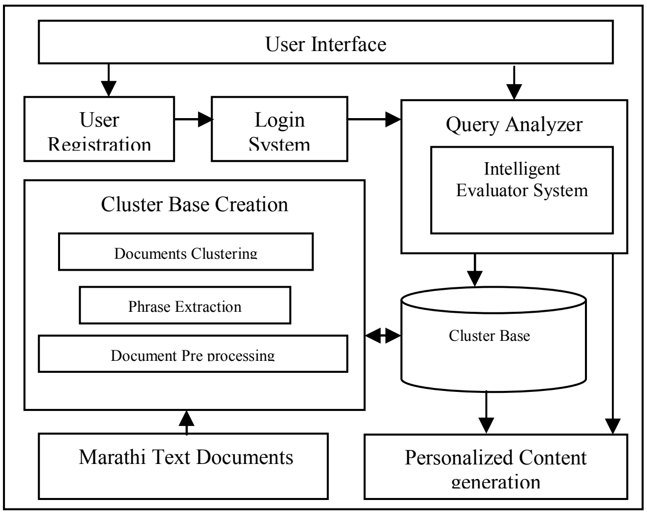

Figure 1. Detail System Architecture Diagram

The system is having two main modules.

The first module involves:

i. The collection of the data set and
ii. Creation of the Cluster base.

The Cluster base is created using Clustering technique of text categorization. Lingo categorization technique is applied for clustering of given Marathi documents' to form the class of a cluster.

The second module involves:

i. Analyze the interests of the particular user's and then
ii. Creating the profile for the identified user's interests.
iii. The query analyzer is created to analyze the user's query and then profile and
iv. Assess the user's choice by checking their browsing history.
v. The personalized content will be generated from the browsing history of the user.

B. An Overview of Processing Query and Clustering:

Generally a user fires their query using a special Web based interface. Then the request of query is sent to a search engine via the API and then a set of relevant documents (top *q*) are retrieved. These *q* relevant documents are given to the clustering process. That is clustering is applied to these query relevant documents only instead of whole documents so that it saves time. These *q* documents are applied to the pre processing phase then pre processed documents are given to the clustering phase and at last documents in the form of cluster are displayed to the user as a result. Figure 2 shows an overview of processing of query using clustering.

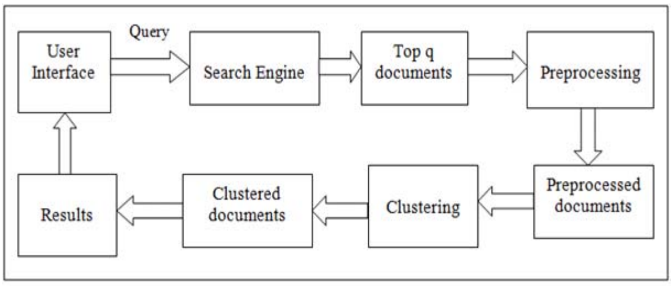

Figure 2 An Over view of query processing using clustering

***Pre processing of Documents**:*

The document pre processing phase removes the words those are useless for the clustering of the documents and so it reduces the clustering process time. Document Pre processing involves two steps:

i) Removal Stop word and
ii) Word Stemming.

Stop word removal: System maintains the list of stop words in order to remove the stop words from the documents. These stop words in the list are compared with the each and every word in the document and if it is matched then removed from the document to improve the efficiency of the clustering. Then each document is converted into feature vector. For English language such lists' can be easily obtained from the Internet, in case of other languages it is much more difficult. For Marathi language stop words' have been obtained by referring dictionary and Marathi documents manually and have been given as input to the system . There are 400 to 500 types of stop words in the English language such as "so", "or", "of", "and", "the," etc. and in the Marathi there are 900 to 1000 stop words such as "Aani", "Athava", "Mhanun", "Kinva" etc. , that does not present useful information about the document's label. These methods reduce the size of the indexing to the great extent.

Stemming: Generally, while searching the documents, user's gives the word of interest in a query, but only variants of that word are available in the document collection. Stemming techniques are applied to convert the text words' into their root word form to improve the efficiency of the search engine in that information retrieval system. A **"stem" or "root word"** is a portion of a word that is left after removal of its affixes. For example the word "processing" must be transformed to the word "process".

For the English language a variety of stemming algorithms are available, Porter stemmer [9] is the most commonly used stemmer. For Marathi language stemmer is available at IIT website and changes are made as per requirement of Marathi words.

Stanislaw Osinski et al. [6], [7] used the LINGO clustering algorithm for the classification of the English and Polish Web documents. They found that 70–80% clusters are useful to the users and 80–95% of snippets into these clusters are matching their query.

Hence from the literature survey it is observed that the Vector space model outperforms the probabilistic model and decided to use LINGO clustering algorithm based on vector space model for the present work.

V. PERFORMANCE EVALUATION

A. Experimental setup

Data Set: Total 107 Marathi documents for the 3 categories: Maharashtra Tourism, Maharashtra festivals and health's programme have been collected from the below websites':

1. www.Maharashtratourism.gov.in
2. www.Maharashtratourism.com
3. www.Maharashtratourism.net
4. www.nic.ac.in

On these websites documents are available in the English language. These documents have been converted into the Marathi language using English to Marathi converter of Indian Language Technology Proliferation and Deployment Centre.

The Lingo algorithm is available on Carrot^2 framework for the English and Polish documents. Changes are made for Marathi documents.

For the classification of 107 Marathi documents of 3 categories, Maharashtra tourism, Marathi festivals and health's programme LINGO clustering algorithm have been used.

C. Performance Metrics

The precision and the recall have been used as measures of the performance for testing the system on the given document set.

Let us explain these performance measures: Consider Q is *a set of* documents, with respect to the user's query, *R is a set* of returned resultant documents. And, let *L* denote the set of all documents in the Q that are actually relevant to the user query. Finally, let L_R be the intersection of *L* and *R*.

Precision is calculated using formula 1:

$$precision = |\, L_R \,| / |R|. \qquad (1)$$

Recall is calculated using formula 2:

$$recall = |\, L_R \,| \,/\, |\, L\, | \qquad (2)$$

F-measure is a combined measure of precision and recall and calculated by using following formula 3:

*F-measure = (2 * Precision * Recall) / (Recall + Precision) (3)*

Preferably, both of the measures must be equal to 1.0.

D. *Evaluation and Results*

The recall, precision and f-measure have been calculated for each category. Tourism data set is tested for queries such as “Shivaji”, “Kolhapur”, “Sant Sai Baba”, “Shani Shingnapur” and “S. S. Deshmukh”, the average precision obtained for such queries is 75.53% and average Recall is 90%. Health programme data set is tested for queries such as “Aarogya”, “Aarogya niyatran”, “Niyantran”, “Rashtriya Karyakram”, average precision obtained is 87.68% and average recall is 100%. Maharashtra Festival data set is tested for queries such as “Utsav”, “Dasara”, “Diwali”, “Ramayana”, “Raksha bandhan” and average precision obtained is 96.52% and average recall is 99%.

Table 1 shows the average of all queries for these categories.

TABLE 1 PERFORMANCE EVALUATION RESULTS USING LINGO

Category	**Average Precision in %**	**Average Recall in %**	**Average F-measure in %**
Tourism	75.53	90	82.13
Health Programme	87.68	100	93.44
Maharashtra Festival	96.52	99	97.75
Total Average	86.58	96.33	**91.10**

The result shows that the LINGO clustering algorithm gives a total average of the 91.10% for the Marathi documents. Thus it shows that the LINGO clustering algorithm for categorizing Marathi documents works well and its performance is better in categorizing the Marathi documents.

VI. CONCLUSION AND FUTURE SCOPE

Several authors of the research papers have focused on automatic categorization of text documents using different techniques and in different languages such as English, Tamil, Polish and Arabic etc. The focus is on providing personalized content in Marathi language to the end user based on their interests using Label induction grouping algorithm as a classifier. Comparative analysis shows that Vector space Model (VSM) gives better result than other models such as Boolean model and Probabilistic model.

LINGO algorithm has given total average of 91.10% for the Marathi documents. Thus it shows that the LINGO clustering algorithm for categorizing Marathi documents works well and its performance is better in categorizing the Marathi documents.

The current work is applicable for the health’s programme, Maharashtra tourism and festival documents of the Maharashtra state, but it can be extended to any category of the documents of any language. Also it can be extended for the country data set.

REFERENCES

[1] Kohilavani, S., Mala, T., Geetha, T.V., “Automatic Tamil Content Generation”, IAMA 2009 © *2010 IEEE*, International Conference, Sep 2009.

[2] E. Iniya Nehru, NIC, Chennai, T. Mala, Anna University, Chennai,“Automatic E-Content Generation”, Sep 2009.

[3] El-Kourdi M., Bensaid A. and Rachidi T., “Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm”, Proceedings of COLING 20th Workshop on Computational Approaches to Arabic Script-based Languages, pp. 51-58, August 2004.

[4] Saleh Alsaleem, Shaqra University, Saudi Arabia, “Automated Arabic Text Categorization Using SVM and NB”, International Arab Journal of e-Technology, Vol. 2, No. 2, June 2011.

[5] Claudio Carpineto, Dawid Weiss, Stanisiaw Osin’ski, *and* Giovanni Romano *“*A Survey of Web Clustering Engines”**,** © *2009 ACM*, ACM Computing Surveys, Vol. 41, No. 3, Article 17, July 2009.

[6] Dawid Weiss, and Stanislaw Osi´nski, “A Concept Driven Algorithm for Clustering Search Results”, 1541-1672/05/$20.00 © *2005 IEEE* INTELLIGENT SYSTEMS.

[7] Stanislaw Osinski, Jerzy Stefanowski, and Dawid Weiss, “Lingo: Search Result Clustering Algorithm Based on Singular Value Decomposition”, Institute of Computing Science, Poznan University of Technology, ul. Piotrowo 3A, 60-965 Poznan, Poland, 2006.

[8] Oren Eli Zamir, “Clustering Web Documents- a Phrase Based Method for Grouping Search Engine Results”, a dissertation submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy University of Washington 1999.

[9] Stanis law Osi´nski, “An Algorithm for Clustering of Web Search Results”, Masters’ thesis, Pozna´n University of Technology, Poland, 2003.

[10] Durgesh K. Shrivastava, “Data Classification Using Support Vector Machine”, Journal of Theoretical and Applied IT © 2005-2009.

[11] XindongWu, J. Ross Quinlan, Vipin Kumar, Joydeep Ghosh, Geoffrey J. McLachlan, Qiang Yang, Hiroshi Motoda, Angus Ng, Philip S. Yu, Bing Liu, “Top 10 algorithms in data mining”, Knowl Inf Syst (2008) 14:1–37 DOI 10.1007/s10115-007-0114-2, © Springer-Verlag London Limited 2007.

[12] Susan Dumais, John Platt, David Heckerman, “Inductive Learning Algorithms and Representations for Text Categorization”, Microsoft Research One Microsoft Way Redmond, WA 98052.

[13] Alex A. Freitas, “A Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery”, Postgraduate Program in Computer Science, Pontificia Universidade Catolica do Parana Rua Imaculada Conceicao, 1155. Curitiba - PR. 80215-901. Brazil.

[14] Fabrizio Sebastiani, “A Tutorial on Automated Text Categorisation”, Istituto di Elaborazione dell’Informazione Consiglio Nazionale delle Ricerche Via S. Maria, 46 - 56126 Pisa (Italy).

[15] Chinglai Hor, Peter A. Crossley, Dean L. Millar, “Application of Genetic Algorithm and Rough Set Theory for Knowledge Extraction”, 978-1-4244-2190-9/07/$25.00 ©*2007 IEEE.*

[16] Oren Zamir and Oren Etzioni, “Grouper: a dynamic clustering interface to Web search results”, Computer Networks (Amsterdam, Netherlands: 1999), 31(11–16): 1361–1374, 1999.

[17] Dell Zhang, Yisheng Dong, “Semantic Online Hierarchical Clustering of Web Search Results”, *Springer* 2004.