ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

An Approach to Bodo Document Clustering

Abdul Hannan[1], Samiran Raj Boro[2], Jay Prakash Sarma[3] Shikhar Kumar Sarma[4]

Research Scholar, Department of Information Technology, Gauhati University, India[1]

U.G. Student, Department of Information Technology, Gauhati University, Guwahati, India [2, 3]

Head, Department of Information Technology, Gauhati University, Guwahati, India[4]

ABSTRACT: Searching a document from the huge collection all over the internet is becoming a challenge. Like other languages Bodo language also providing content to the electronic world. Bodo is widely used in the North Eastern states of India. As text documents are increasing exponentially across the web, grouping similar documents for versatile applications suffers from challenges in dealing with problems of high dimensionality, scalability, accuracy and meaningful cluster label. Clustering of documents has many effective points in information retrieval and data mining field. However, in this presentation a practical approach for clustering of Bodo documents are implemented in order to find suitable clustering technique specifically for this purpose.

KEYWORDS: Document clustering, Bodo language, NLP, Data Mining, Stammer

I. INTRODUCTION

Bodo language is one of the 2 scheduled languages that is given a special constitutional status in India. Bodo language uses Devnagari script for writing script which is also used by Hindi language, although it also has a long history of using the Latin script and Assamese script. The use of Devanagiri in Bodo Language just completed 50 years on 2013. It is closely related to the Dimasa language of Assam, the Garo language of Meghalaya and also related to the language of Kokborok language spoken in Tripura. The Bodo speaking areas of Assam at present stretch from Dhubri in the west to Sadiya in the east.[1]

In Jalpaiguri and other adjacent districts of Bengal, the Bodos are known as "Mech". The Bodo language was introduced as a medium of instruction in the primary school in Bodo dominated areas in 1963. In Assam, Bodo dominated areas is commonly known as Boroland.

Clustering or cluster analysis is a task of grouping similar (in some sense or another) set of object. It involves task of dividing data points into homogeneous classes or clusters. Document clustering (or Text clustering) is automatic document organization, topic extraction and fast information retrieval or filtering. Its main motive is to divide a collection of text documents into different category groups so that documents in the same category group describe the same topic, such as classic music, history or romantic story.

Classification is similar to clustering in that it also segments data into distinct segments called classes. But unlike clustering, a classification analysis requires that the end-user/analyst know ahead of time how classes are defined.

II. REVIEW OF LITERATURE

Till date there is no known works for Document clustering in Bodo language. But with the recent research done in Natural Language Processing (NLP) across various major institutes in India like Indian Institute of Technology Bombay, Gauhati University, Tezpur University Bodo language has also been digitalized for localization. Hence some resources like Corpus, Word-net, Suffix list etc. are developed.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

Indo Wordnet developed at India Institute of Technology Bombay It is a linked lexical knowledge base of Bodo language wordnets, where a word can be looked up for getting list of synonyms, antonym etc. Applications like spell checker are developed at Gauhati University. Moreover great number of corpus is generated by digitalizing offline resources like articles, newspapers, books etc.

Clustering techniques are not new to us. k-means was first used by James MacQueen in 1967, though the idea goes back to Hugo Steinhaus in 1957. The standard algorithm was first proposed by Stuart Lloyd in 1957 as a technique for pulse-code modulation, though it wasn't published outside Bell labs until 1982. In 1965, E.W.Forgy published essentially the same method, which is why it is sometimes referred to as Lloyd-Forgy, too. A more efficient version was proposed and published in Fortran by Hartigan and Wong in 1975/1979.

In recent years considerable effort has been put into improving algorithm performance of the existing algorithms. With the recent need to process larger and larger data sets (also known as big data), the willingness to treat semantic meaning of the generated clusters for performance has been increasing. This led to the development of pre-clustering methods such as canopy clustering, which can process huge data sets efficiently, but the resulting "clusters" are merely a rough pre-partitioning of the data set to then analyse the partitions with existing slower methods such as k-means clustering.

Jain et al.[10] covered the topic very well from the point of view of cluster analysis theory and they break down the methodologies mainly into partition and hierarchical clustering methods. In this work the clustering algorithm used falls under the partition category, it also discusses various types of clustering methods and categorizes them into partitioning, geometric embedding and probabilistic approaches.

Text mining research in general relies on a vector space model, first proposed by Salton [11] to model text documents as vectors in the feature space. Features are considered to be the words in the document collection and feature values come from different term weighting schemes, the most popular of which is the Term Frequency-Inverse Document Frequency (tfidf) term weighting scheme. This model is simple but assumes independence between words in a document, which is not a major problem for statistical-based methods, but poses difficulty in phrase-based analysis.

A relatively new technique was proposed by Zamir and Etzioni[12] in 1998. They introduced the notion of phrase-based document clustering. They used a generalized suffix-tree to obtain information about the phrases and used them to cluster the documents.

Sun et al. [13] in 2008 developed a novel hierarchical algorithm for document clustering. They used cluster overlapping phenomenon to design cluster merging criteria. The system computes the overlap rate in order to improve time efficiency and the veracity and the line passed through the two cluster's center instead of the ridge curve. Based on the hierarchical clustering method it used the Expectation-Maximization (EM) algorithm in the Gaussian mixture model to count the parameters and make the two sub-clusters combined when their overlap is the largest.

III. NATURAL LANGUAGE PROCESSING

Natural Language Processing concerned with the interactions between computers and human (natural) languages. Many challenges in NLP involve natural language understanding – that is, enabling computers to derive meaning from human or natural language input. For our purpose we need to process the corpus by detagging, removing stop words and stemming. The corpus was downloaded from Technology Development for Indian Languages website from [9]

1. DETAGGER

   Detagger is a program which finds the special tags in document and filters it away. Tags which need to be filtered in Bodo language are: , " . | ? > < - ! ( ) { } / etc.
   It's a simple process of search and replace. Regular expressions may be used for faster process.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

Example: आबहावायाव मोननै गाहाइ थादेरसाफोर थायो। बेफोर जादों- देहायारि आरो जिवारि। बार, दै, हा।

After detagging: आबहावायाव मोननै गाहाइ थादेरसाफोर थायो बेफोर जादों देहायारि आरो जिवारि बार दै हा

2. TOKENIZER

Tokenization is the process of breaking a stream of text up into words, phrases, symbols, or other meaningful elements called tokens. For Bodo language processing tokenization must be done in word level (separated by space).

Example: {आबहावायाव} {मोननै} {गाहाइ} {थादेरसाफोर} {थायो} {बेफोर} {जादों} {देहायारि} {आरो} {जिवारि} {बार} {दै} {हा}

3. STOP WORDS REMOVAL

Stop words are words which are filtered out prior to, or after, processing of natural language data (text). Any group of words can be chosen as the stop words for a given purpose.

For our project purpose, we identified a list of 172 stop words. These include inflected words. Some of them are: आरो ए एबा ऐ जेरै नों सोर बबे etc.

Inflected words were also listed in our stop words list to reduce computing time, or else we must perform stemming before removing the stop words.

4. STEMMING

Stemming is the process for reducing inflected (or sometimes derived) words to their stem, base or root form. The stem need not be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root.

Different approach can be used for stemming:

- **Lookup algorithm:** A simple stemmer looks up the inflected form in a lookup table. The advantages of this approach is that it is simple, fast, and easily handles exceptions. The disadvantages are that all inflected forms must be explicitly listed in the table.
- **Prefix suffix stripping algorithm:** A smaller list of "rules" is stored which provides a path for the algorithm, given an input word form, to find its root form.
  A good morphological analysis is needed to create prefix suffix stripping algorithms. Porter stemmer, Lovins Stemmer, Paice/Husk stemmer[2] are of such type. Developing such kind of algorithm are beyond the scope of this project as we are not language expert and there are great numbers of rules unknown to us.
  Some of the problems to create a stemmer using morphological analysis in Bodo language are discussed below:
- **Multiple suffix:** A single word can have multiple suffix appended to it. Example: मानसिनि मानसिनिफ्राय सोरबाफोर सोरबाफोरनि
- Irregular inflection: Generally Bodo words follow simple inflection rules like गोसोआ = गोसो + आ but some are completely irregular like बिमाया = बिमा + आ, थुनलाया = थुनलाइ + या [3]

The approach we used in our project was combination of Look up algorithm and suffix striping. We identified 206 suffixes for our project work. A word list of 6298 was used. The word list was collected from the Department of IT, Gauhati University.

This algorithm has a success rate of 58.34% with the current word list and suffix list. This is not a good number but this will increase relatively with the increase of word list and suffix list.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

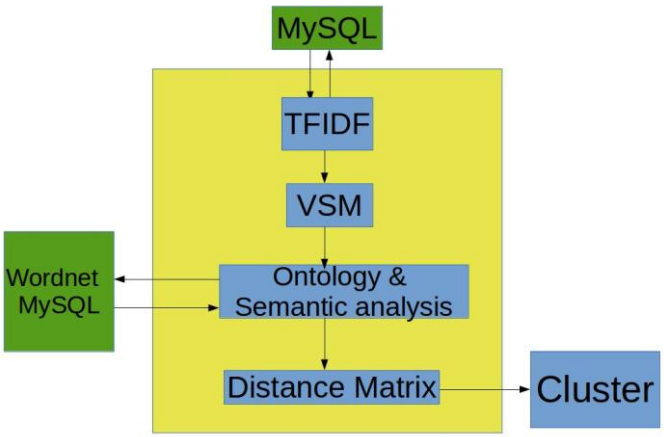

Fig. 1 Clustering process overview

IV. DOCUMENT CLUSTERING

There are many clustering techniques currently in use. Most of them are variants from previously known algorithms. But document clustering have its speciality then any other clustering as documents are non-numeric data. The common technique for documents clustering are k-means and hierarchical clustering.[4] Many other algorithms are derived from these algorithms. Although the complexity of hierarchical clustering is more than that of k-means, hierarchical clustering gives precise results due to its comparison nature.

K-means is one of the simplest unsupervised learning algorithms. It aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean. K-Means clustering generates a specific number of disjoint, flat (non-hierarchical) clusters. It is well suited to generating globular clusters. The K-Means method is numerical, unsupervised, non-deterministic and iterative.[5]

Given a set of observations (x_1 , x_2 , ..., x_n), where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into k sets ($k \leq n$) $S = S_1$, S_2 , ..., S_k so as to minimize the within-cluster sum of squares (WCSS):

$$J = \sum_{i=1}^{k} \sum_{j=1}^{n} |x_j - \mu_i|^2$$

where μ_i is the mean of points in S_i.

Hierarchical clustering is a method of cluster analysis which seeks to build a hierarchy of clusters. Thus, a structure that is more informative than the unstructured set of clusters. Hierarchical clustering does not require pre-specifying the number of cluster. These advantages of hierarchical clustering come at the cost of lower efficiency. In general, the merges and splits are determined in a greedy manner. The results of hierarchical clustering are usually presented in a dendrogram.[6]

There are two types of hierarchical clustering:

- **Agglomerative**: Agglomerative hierarchical clustering is a bottom-up clustering method where clusters have sub-clusters, which in turn have sub-clusters, etc. Agglomerative hierarchical clustering starts with every single object (or sample) in a single cluster. Then, in each successive iteration, it agglomerates (merges) the closest pair of clusters by satisfying some similarity criteria, until all of the data is in one cluster.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

- **Divisive**: A top-down clustering method and is less commonly used. It works in a similar way to agglomerative clustering but in the opposite direction. This method starts with a single cluster containing all objects, and then successively splits resulting clusters until only clusters of individual objects remain.

After cleaning the corpus in the NLP stage, it needs to pass through various steps. The clustering process is shown in Figure 2. We have used MySql for storing.

A. Text Term Frequency Inverse Document Frequency (TF-IDF)

1) Term Frequency: It is simple number of occurrences of the term in the document. We used the below mentioned equation to compute Term Frequency:

$$tf(t,d) = \frac{f(t,d)}{|\omega \in d|}$$

tf(t,d) is the term frequency
f(t,d) is the frequency of the term 't' in document 'd'
$|\omega \in d|$ is the total number of words in the document'd'

2) Inverse Document Frequency: It is a measure of whether the term is common or rare across all documents. Raw term frequency suffers from a critical problem: all terms are considered equally important when it comes to assessing relevancy on a query. To this end, a mechanism for attenuating the effect of terms that occur too often in the collection to be meaningful for relevance determination is required. It is obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient.[7]

$$idf(t,D) = \log\left(\frac{|D|}{|(d \in D : t \in d)|}\right)$$

|D| is the total number of documents in the corpus
$|(d \in D : t \in d)|$ is the number of documents where the term 't' appears

3) *TF-IDF*: Thus TF-IDF is calculated as:

tfidf(t,d,D) = tf(t,d) x idf(t,D)

B. Vector Space Modelling

Vector space model is an algebraic model for representing text documents (and any objects, in general) as vectors of identifiers. A text document is represented as a vector of terms $<t_1, t_2, ..., t_i, ..., t_m>$. Each term ti represents a word. A set of documents are represented as a set of vectors, that can be written as a matrix. Where each row represents a document, each column indicates a term, and each element x_{ji} represents the frequency of the i^{th} term in the j^{th} document.

$$\begin{pmatrix} x_{11} & \cdots & x_{1i} & \cdots & x_{1m} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{j1} & \cdots & x_{ji} & \cdots & x_{jm} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & \cdots & x_{ni} & \cdots & x_{nm} \end{pmatrix}$$

Fig. 2 Vector Space Model matrix representation

C. Ontology and Semantic Enhancement (NLP step)

Ontology formally represents knowledge as a set of concepts within a domain, and the relationships between pairs of concepts. Each element of a document vector considering ontology is represented by:

$$\tilde{x}_{ji_1} = x_{ji_1} + \sum_{i_2=1, i_2 \neq i_1}^{m} \delta_{i_1 i_2} x_{ji_2}$$

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

Where x_{ji1} is the original frequency of t_{i1} term in the j^{th} document, δ_{i1i2} is the semantic similarity between t_{i1} term and t_{i2} term.

WordNet can be used for such enhancement. But the WordNet we have (received from Department of IT, Gauhati University) does not have semantic similarity values. It has sets of synonyms. So, our formula was simple to add all the values of synonyms of the same set.

D. Distance Matrix

Distance Matrix is created from VSM. It is needed to compare the similarity or dissimilarity between documents. It is a square matrix with n x n size, where n is the total number of document in the corpus.

Distance matrix is generated in many ways, like by using Euclidean distance, Manhattan distance, cosine distance etc. We used Euclidean distance to form the distance matrix.

Euclidean distance:

$$|a-b|_2 = \sqrt{\sum_i (a_i - b_i)^2}$$

E. Single link Agglomerative Hierarchical Clustering

In single-link clustering or single-linkage clustering, the similarity of two clusters is the similarity of their most similar members. We pay attention solely to the area where the two clusters come closest to each other. Other, more distant parts of the cluster and the clusters' overall structure are not taken into account.

F. Plotting the cluster

The hierarchical cluster is plotted using heat-maps and dendogram.

1) Heat Map: The easiest way to understand a heat map is to think of a table or spreadsheet which contains colors instead of numbers. The default color gradient sets the lowest value in the heat map to dark blue, the highest value to a bright red, and mid-range values to light gray, with a corresponding transition (or gradient) between these extremes.[8] Heat maps are well-suited for visualizing large amounts of multi-dimensional data and can be used to identify clusters of rows with similar values, as these are displayed as areas of similar colour.

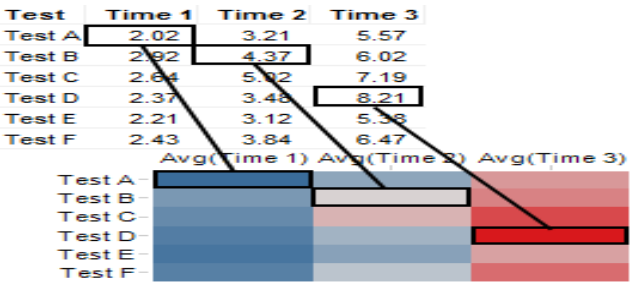

Fig. 3 Heat Map cells

2) Dendograms: It is often useful to combine heat maps with hierarchical clustering, which is a way of arranging items in a hierarchy based on the distance or similarity between them. The result of a clustering calculation is presented either as the distance or the similarity between the clustered items depending on the selected distance measure. The result of a hierarchical clustering calculation is displayed in a heat map as a dendrogram, which is a tree-structure of the hierarchy. Row dendrograms show the distance (or similarity) between rows and which nodes each row belongs to as a result of the clustering calculation. Column dendrograms show the distance (or similarity) between the variables (the selected cell value columns).

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

G. Cluster Analysis

Sports news items were selected for clustering. 5, 10, 20 documents were clustered with minimum support of 20%, 30%, 40% each and number of keywords are recorded.

TABLE I: CLUSTER ANALYSIS (KEYWORD DISCOVERED)

No. of Doc.	Minimum Support (in percentage)		
	20	**30**	**40**
5	86	28	25
10	73	27	16
20	139	46	32

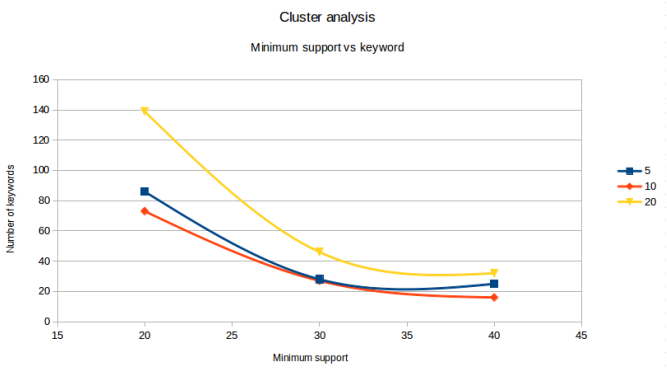

Fig. 4 Cluster analysis: Minimun support vs number of keywords

A minimum support vs number of keyword graph is generated in figure 5. From this figure we can see that the number of keywords is high for lower minimum support and low for higher number minimum support. As the minimum support increases the number of keywords is consistence. Thus, choosing a minimum support between 30% and 35% is recommended.

A heat map with row (document) dendogram and column (keyword) dendogram is generated like in figure 7. This was generated by using 5 document with 20% support.

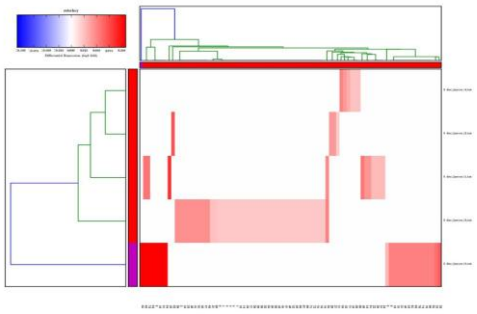

Fig. 5: Cluster: 5 document with 20$ minimum support

V. CONCLUSION

The derived work from this project can be used in many real life applications. One such is for creating Bodo language search engine. Other advantage of document clustering like information retrieval, knowledge discovery are also notable.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

Term Frequency Inverse Document Frequency (TF-IDF) was used to weight the terms as it removes the affect of terms that occur too often. Euclidean metric was used to create the distance matrix; other approach may also be used.
For further enhancements of this project the Bodo dictionary must be enhanced to list every possible words since the current stemmer success rate is 58.34%. Suffix list also affects the outcome of the stemmer, so a better suffix list must also be developed. A better stemmer algorithm for Bodo can be designed by using morphological analyser much like the Porter's algorithm in order to achieve an optimal stemmer success rate.

The execution time of our program can be reduced using multi-threading for the processing of each document in the corpus and inserted into MySQL table. Use of NoSQL like Apache Cassandra can play an important for faster processing.

REFERENCES

[1] Wikipedia. Bodo language. Retrieved March, 2013. [Online]. Available: http://en.wikipedia.org/wiki/Bodo_language
[2] Hooper, R., & Paice, C. (2005). The Lancaster stemming algorithm. Retrieved April, 2013. [Online[Available: http://www.comp.lancs.ac.uk/computing/research/ stemming/index.htm
[3] Aalin Brahma. Boro banankhanti.
[4] Anil K. Jain, and Richard C. Dubes. Algorithms for clustering data. Prentice-Hall, Inc., 1988.
[5] Improved outcomes software. K-means clustering overview, Retrieved March, 2013. [Online]. Available: http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/K-Means_Clustering_Overview.htm
[6] Improved outcomes software. Agglomerative hierarchical clustering overview, Retrieved March, 2013. [Online]. Available: http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/Agglomerative_Hierarchical_Clustering_Overview.htm
[7] Christopher D. Manning,, Prabhakar Raghavan, and Hinrich Schütze. Introduction to information retrieval. Vol. 1. Cambridge: Cambridge University Press, 2008.
[8] Spotfire technology network. Heat map, Retrieved March, 2013. [Online]. Available: http://stn.spotfire.com/spotfire_client_help/heat/heat_what_is_a_heat_map.htm
[9] Indian Language Technology and Deployment Centre. Bodo document corpus, Retrieved April, 2013. [Online]. Available: http://tdil-dc.in/index.php?option=com_up-download&task=view-download-tool&view=GET&toolid=451
[10] Anil K. Jain, M. Narasimha Murty, and Patrick J. Flynn. "Data clustering: a review." ACM computing surveys (CSUR) 31.3 (1999): 264-323.
[11] Gerard Salton. "The SMART retrieval system—experiments in automatic document processing." (1971).
[12] Oren Zamir and Oren Etzioni. "Web document clustering: A feasibility demonstration." Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1998.
[13] Haojun Sun, Zhihui Liu, and Lingjun Kong. "A document clustering method based on hierarchical algorithm with model clustering." Advanced Information Networking and Applications-Workshops, 2008. AINAW 2008. 22nd International Conference on. IEEE, 2008.