Research Journal of Applied Sciences 8 (7): 358-368, 2013
ISSN: 1815-932X

Integrating Ontology to Enhance HCL-Based Text Document Clustering

[1]S. Vijayalakshmi and [2]D. Manimegalai
[1]Department of Applied Sciences, Sethu Institute of Technology, Kariapatty, India
[2]Department of Information Technology, National Engineering College, Kovilpatty, India

Abstract: Increasingly large text datasets and the high dimensionality associated with natural language is a great challenge of text mining. Initially, researchers have been compared using three types of Document Representation (Bag of Word (BoW), Bag of Noun (BoN) and Bag of Phrase (BoP)) and researchers found that Bag of Noun and Bag of Phrase are performing better than BoW. BoP significantly improves the better F-measure than BoN and BoW when the corpus is smaller. If the corpus is larger, it increases the dimensionality. BoN document representation working efficiently and also used to reduce its dimensionality when the corpus is larger in text document clustering than BoP and BoN. Researchers have been used Bag of Noun document representation. Nouns are checked with ontology and extracted to construct term document matrix, although it reduces the dimension and gives semantics. The comparative study result shows that the performance of Bag of Noun document representation is better than Bag of Phrase. Exploration of learning algorithm gives promising results in recent years. In this study, researchers propose ontology based OHCLK-Means Clustering algorithm. It significantly improves the clustering quality than ontology based K-means and ontology based ONVK-means.

Key words: BON, BOP, cosine similarity, ontology based K-Means, RiTa WordNet, ontology based NVK-means, ontology based HCLK-means

INTRODUCTION

Today, constant and rapid changes in information and communication technologies offer ubiquitous access to vast amounts of information and make an exponential increase of the amount of documents available online. While more and more textual information is available electronically, effective retrieval and mining is getting more and more impossible without efficient organization, summarization and indexing of document content. Among different approach tackled this problem, document clustering is one of the main and enabling approach. In general, clustering or cluster analysis is the process of automatically identifying similar terms to group them together into clusters. Clustering is an unsupervised learning method which means no labelled training examples need to be supplied for the clustering to be successful. In other words, no output data is necessary.

For simplicity sake, researchers want to make optimal decisions under uncertainty by calibrating parameters based on available resources. Neural network gives the perception of an intelligent system that can adapt to change by utilizing only the available resources. Researchers are exploring different ways of extending the ideas behind the linear Processing Element (PE) or ADALINE (Guerrero-Bote *et al*., 2003) (for adaptive linear element) by introducing new biological concepts such as those described in Competitive Learning and Hebbian Learning which allow us to make optimal decisions under uncertainty.

Learning (Laia and Tsung-Jen, 2010; Pessiot *et al*., 2010) includes supervised and unsupervised. Supervised learning learn from the outputs by usage of the feedback error (for example, ADALINE) to change the weights. Such systems require a training set, since a desired response is used to guide the learning process. Unsupervised/self-organized-learn from the inputs by applying internal rules to change the weights.

In this study, a systematic study is conducted. Two different document representation methods namely Bag of Phrase/Unigrams and Bag of Nouns are used. Traditional tf-idf weighting with Euclidian distance similarity measures are used in the context of the text clustering problem. The contribution for this research, researchers have been exploited two different features (Noun, Phrase or Unigram) using RiTa WordNet Ontology. These two features selection may implemented by using K-means (Laia and Tsung-Jen, 2010) clustering and hierarchical clustering in few publication. But researchers have incorporated with HCLK-means algorithm and NVK-means algorithm to improve the cluster quality. Researchers have added K-means clustering also to compare the existing clustering results.

Corresponding Author: S. Vijayalakshmi, Department of Applied Sciences, Sethu Institute of Technology, Kariapatty, India

Novel clustering evaluation methods are used and evaluated with several standard benchmark datasets for this application.

TEXT DOCUMENT CLUSTERING

The process of text document clustering contains two major steps namely document representation and dimensionality reduction.

Document representation: Document representation (Konchady, 2007; Shafiei *et al*., 2007) is the process of converting raw documents into easily accessible representation. It maps the content of a document dj and converts into compact representation (Konchady, 2007). In this research, researchers use rapid miner to evaluate the performances of BoW, BoP and BoN. The rapid miner project was started in 2001 by Ralf Klinkenberg, Ingo Mierswa and Simon Fischer at the Artificial Intelligence Unit of the Dortmund University of Technology. Rapid miner is an environment for machine learning, data mining, text mining, etc. It uses learning schemes and attributes evaluators from the weka machine learning environment and statistical modelling schemes from R-project. Rapid miner includes WordNet and POS (Part of Speech) tagger, classifier, etc. The document representation in rapid miner is done by the following steps:

- Processing documents from files is an operator. Choose it from operator window and place it in the process window. It performs pre-processing using transform case, tokenize, stop word, stem operators and generate two vectors which includes wordlist and TF-IDF weight. Researchers can call it BoW representation
- To generate BoP and BoW representation, add filter token (POS tags) operator after stemming process. Filter tokens based on the specified types of POS tags. The possible POS tags are in STTS System for German tagging and in PENN System for English tagging and are defined by a regular expression of types. For example: the expression JJ.*|N.* would keep all adjectives and nouns
- Add X-validation operator to calculate the performance of this representation. It generates precision, recall and accuracy

RiTa WordNet ontology-the hidden semantic web: It is basically a giant lexical database. It has many more capabilities and there are different varieties. RiTa http://www.rednoise.org/rita/wordnet/documentation/index.htmWordNet (Pullwitt, 2002) is used to extract semantic similarities or semantic distances. In Rita WordNet, User can able to extract distances by giving the best Part of Speech (POS) or all the existing POS.

The RiTa toolkit (Howe, 2009) is implemented as a Java library comprised of seven independent packages. The core object collection is comprised of approximately 20 classes within the RiTa package, all of which follow similar naming and usage conventions. The rest of the packages provide support for these core objects but are not directly accessed under typical usage. Each core object defines the basic properties, methods and support structures for a specific task.

RiTokenizer: A simple tokenizer for word and sentence boundaries with regular expression support for customization.

RiStemmer: A simple stemmer (based on the porter algorithm) for extracting roots from words by removing prefixes and suffixes.

RiWordNet: An intuitive interface to the WordNet ontology providing definitions, glosses and a range of onyms (hyponyms, hyponyms, synonyms, antonyms, meronyms, etc.) can be transparently bundled into web-based, browser-executable programs.

RiPosTagger: A light-weight transformation-based part-of-speech tagger based on an optimized version of the Brill algorithm.

Proposed research: In this study, researchers concentrate more in pre-processing and dimension reduction based on its feature or attribute frequency. Here, researchers generate RiTa WordNet based BoP document representation and RiTa WordNet based BoN document representation which is mainly used for dimension reduction.

RiTa WordNet based BOP and BON representation: A phrase (Wei *et al*., 2006) is a group of closely related words that function together as a single element, such as subject, verb, adjective or adverb. BOP stand for bag of phrase (Text representation) (Brill, 1992); a text (a document) is represented as an unordered collection of words, disregarding grammar and even word order. The original word sequence is categorical, high dimensional and sparse. The smoothing method is employed by the bag of phrases representation.

A common alternative to the use of dictionaries is the hashing trick where words are directly mapped to indices with a hashing function. By mapping words to indices directly with a hash function, no memory is required to store a dictionary. Hash collisions are typically dealt with by using freed-up memory to increase the number of hash buckets. In practice, hashing greatly simplifies the implementation of BoP Models and improves their scalability.

RiTa WordNet based BON representation: Automatically extracting POS (Chen *et al*., 2010; Zheng *et al*., 2010) from an unstructured text has been studied since 1960. The task of POS tagging is simply assigning a part of speech to a word. The eight main word classes are verb, noun, adjectives, adverbs, prepositions, conjunctions, pronouns and determiners. POS taggers report precision rates of 90% or higher (Brill, 1992). POS tagging problem (Konchady, 2007) is reduced to extracting the most discriminating feature for a word. Noun is most frequently used word class and more meaningful. Noun based vector representation produce compact vector with better semantics.

The BoP/BoN document vectors (Kashef and Kamel, 2009) contain term frequencies. The simplest approach is to assign the weight to be equal to the number of occurrences of term t in document d. This weighting scheme is referred to as term frequency (Qian and Suen, 2000) and is denoted as $tf_{t,d}$ with the subscripts denoting the term and the document in order. Raw term frequency as above suffers from a critical problem: all terms are considered equally important when it comes to assessing relevancy on a query. In fact certain terms have little or no discriminating power in determining relevance (Fig. 1 and 2).

Document frequency df_t, defined to be the number of documents in the collection that contain a term t. This is because in trying to discriminate between documents for the purpose of scoring it is better to use a document-level statistic (such as the number of documents containing a term) than to use a collection-wide statistic for the term. Denoting as usual the total number of documents in a collection by N, researchers define the inverse document frequency (idf) of a term t as follows:

$$idf_t = \log\left(\frac{N}{df_t}\right) \quad (1)$$

It is common to weight terms by various schemes, the most of popular of which is http://en.wikipedia.org/wiki/Tf%E2%80%93idf. The tf-idf weighting combines the definitions of term frequency and inverse document frequency to produce a composite weight for each term in each document. The tf-idf weighting scheme assigns to term t a weight in document d is given by:

$$tf\text{-}idf_{t,d} = tf_{t,d} \times idf_t \quad (2)$$

In other words, $tf\text{-}idf_{t,d}$ assigns to term t a weight in document d that is:

- Highest when t occurs many times within a small number of documents (thus lending high discriminating power to those documents)

Algorithm to construct RiTa WordNet based BoP
Input: Dataset D with 'm' categories
Ontology: RiTa WordNet

- Output: Bag of phrases
- Receive the text to be parsed from the dataset
- Build a custom stop word list based on the text
- Add RiTa Word net ontology
- If received text is phrase (Noun | verb | adjective | adverb) from RiTa word net then
    - Call BOP to generate a list of tokens from the text
    - Initialize a list of words and loop through the list of tokens
        - Skip token that are <3 characters long
        - Skip tokens that are found in stopword list.
    - Add the tokens to the list
- Return the list called bag of phrase
- For all BoP do
    - Find the frequency of relevant term P>1
    - Then estimate the goodness of term
    - Else discard

  End.
- Apply a sorting and select top 'q' features

Fig. 1: RiTa WordNet based BoP document vector construction algorithm

Algorithm to construct RiTa WordNet based BoN
Input: Dataset D with 'm' categories
Ontology: RiTa WordNet
Output: Bag of Nouns

- Receive the text to be parsed from the dataset
- Build a custom stop word list based on the text
- If received text is Noun using RiTa WordNet ontology then
  - Call BOP to generate a list of tokens from the text
  - Initialize a list of words and loop through the list of tokens
    - Skip token that are <3 characters long
    - Skip tokens that are found in stop word list.
  - Add the tokens to the list
- Return the list called bag of Noun
- For all BoN do
    - Find the frequency of relevant term N>1
    - Then estimate the goodness of term
    - Else discard

End.

Fig. 2: RiTa WordNet based BoN document vector construction algorithm

Algorithm: Ontology based K-Means clustering
Input: weighted RiTa WordNet based BoP/construct RiTa WordNet based BoN document vector, number of cluster.
Output: Set of K clusters $Y = \{Y_1, Y_2, \ldots Y_n\}$ $y_n \in \{1, 2, \ldots k\}$
Step1: Initialization: $Y = \{\}$, Randomly, select k initial centroid vector $\{\mu_1, \mu_2, \ldots \mu_k\}$.
Step 2: For each data vector x_n at iteration t, using the minimum-distance Euclidean criterion

$$y_n \quad \min_k \| x-\mu_k(t) \|$$

Step 3: t = t+1
Step 4: If no noticeable changes occur then stop, otherwise go back to Step 2 and continue until the minimum-distance Euclidean criterion is satisfied.

Fig. 3: Ontology based K-Means clustering algorithm

- Lower when the term occurs fewer times in a document or occurs in many documents (thus offering a less pronounced relevance signal)
- Lowest when the term occurs in virtually all documents
- In document vector, researchers may view each document as a vector with one component corresponding to each term in the dictionary together with a weight for each component. For dictionary terms (BoP/BoN) doesn't occur in a document then the weight is zero. This vector form will prove to be crucial to scoring and ranking

Ontology based K-means clustering algorithm: BoN and BoP document representation is constructed with the help of RiTa WordNet ontology and the term weight ($tf\text{-}idf_{t,d}$) is calculated by using the (Eq. 2) traditional method. The attributes or instances are Nouns in BoN. Noun, Verb, Adjectives, Adverb are instance in BoP. These attributes or terms or instances are extracted with help of RiTa WordNet. Resultant weighted term document matrix is used for K-mean clustering. So, hereafter researchers call it as ontology based K-mean clustering (Fig. 3).

K-means (Zhong, 2005; Michael, 2000) is a partitional clustering algorithm. Let the set of data points (or instances) D be $\{x_1, x_2, \ldots, x_n\}$, where $x_i = \{x_{i1}, x_{i2}, \ldots, x_{ir}\}$ a vector in a real-valued space $X \subseteq R_r$ and r is the number of attributes (dimensions) in the data. The k-means algorithm partitions the given data into k clusters. Each cluster has a cluster center called centroid. k is specified

Algorithm: Ontology based NVK-Means
Input Vector: weighted RiTa WordNet based BoP / RiTa WordNet based BoN document vector $X = \{x_1, x_2, \ldots, x_n\}$, number of cluster.
Output: Set of K clusters $Y = \{y_1, y_2, \ldots y_n\}$ $y_n \in \{1, 2, \ldots K\}$
Step1: Initialization: $Y = \{\}$, Randomly, select k initial centroids μ_i, for $i = 0, 1, \ldots k$ for K cluster y_n

Step 2: New centroids $\mu_i = 0, 1, \ldots, k$ are calculated as $\mu_k = \sum_{x \in x_k} \frac{X}{\left\| \sum_{x \in x_k} X \right\|}$

Step 3: Find and update the best matching centroid j_X at iteration t, using the minimum-distance Euclidean criterion

$$y_n \quad \min_k \left\| x - \mu_k(t) \right\| = \left\{ \sum_{i=1}^{n} \left[x_i - \mu_k(t) \right]^2 \right\}^{\frac{1}{2}}$$

Step 4: t = t+1
Step 5: If no noticeable changes occur then stop, otherwise go back to Step 2 and continue until the minimum-distance Euclidean criterion is satisfied.

Fig. 4: Ontology based NVK-means clustering algorithm

by the user. The objective of standard k-means clustering (Kashef and Kamel, 2009; Qian and Suen, 2000) is to minimize the mean-squared error:

$$E = \frac{1}{N} \sum_{x} \|X - \mu_k\|^2 \quad (3)$$

where, $k(x) \arg = \min_{k \in \{1\text{-}K\}} \|X\text{-}\mu_k\|$ is the index of the closest cluster centroid to x, N is the total number of data vectors. The k-means algorithm (Zheng *et al*., 2010; Aliguliyev, 2009) can be used for any application dataset where the mean can be defined and computed. In Euclidean space, the mean (μ) of a cluster is computed with:

$$\mu = \frac{1}{|C_k|} \sum_{x_i \in c_k} x_i \quad (4)$$

where, $|C_k|$ is the number of data points in cluster C_k. The distance from one data point x_i to a mean (centroid) μ_k is computed with.

ONTOLOGY BASED NVK-MEANS CLUSTERING ALGORITHM

Unit vector in a normed vector space (Zhong, 2005) is a vector whose length is 1 (the unit length), sometimes also called a direction vector. A unit vector is often denoted by a lowercase letter with a hat, like this: $\widehat{NV}$ the unit vector $\widehat{NV}$ having the same direction as a given (nonzero) vector $\widehat{NV}$ defined by:

$$\widehat{NV} = \frac{X}{\|X\|} \quad (5)$$

The term "norm" is often used without additional qualification to refer to a particular type of norm (such as a matrix norm or vector norm). Most commonly, the unqualified term "norm" refers to the flavour of vector norm technically known as the L2-norm. This norm is variously denoted $\|X\|$ and gives the length of an n-vector $X = (x_1, x_2, ..., x_n)$. The main difference from standard k-means is that the re-estimated mean vectors need to be normalized to unit-length. Let (μ), i = {1, 2, ..., k} be a set of unit-length centroid vectors. NVK-mean again iterates between a data assignment step and a mean estimation in Fig. 4.

ONTOLOGY BASED HCLK-MEANS CLUSTERING ALGORITHM

Hard competitive learning comprises methods where each inputs only determines the adaptation of one unit, the winner. There are two specific methods for updating like batch or online. In batch methods, all possible inputs are evaluated first before any adaptation are done. This is iterated number of times. In online methods (e.g., K-means). Perform an update directly after each inputs. The main problem of HCL is that different random initializations may lead to very different results.

Unsupervised learning (Zhong, 2005) algorithms aim to learn rapidly and can be used in real-time. Competitive

Algorithm: Ontology based HCLK-Means clustering Algorithm
Input Vector: Weighted RiTa WordNet based BoP / RiTa WordNet based BoN Normalized document vector $X = \{x_1, x_2, \ldots, x_n\}$, number of cluster K and learning rate (= 0.01.
Output: Set of K clusters, Vector $Y = \{Y_1, Y_2, \ldots Y_n\}$, $Y_n = \{1, 2, \ldots K\}$
Step 1: Initialization: $Y = \{\}$, Randomly, select k initial centroids vector μ_i, for $i = 0, 1, \ldots k$ for K cluster Y_n, $t = 0$, assign a small positive value to the learning rate parameter (= 0.01.
Step 2: For m = 1 to M
For n = 1 to N
Step 2.1: For each data vector x_n, find the best matching centroid using the minimum-distance Euclidean criterion $y_n = \underset{k}{\operatorname{argmin}} \|X-\mu_k\|$
Step 2.2: Update centroid $\mu_{yn} = \dfrac{\mu_{yn}+\alpha^{(t)}(x-\mu_{yn})}{\left\|\mu_{yn}+\alpha^{(t)}(x-\mu_{yn})\right\|}$
Step 2.3: $t = t+1$

Fig. 5: Ontology based HCLK-means clustering algorithm

learning (Zhang *et al*., 2010; Zhong, 2005) "norm" is an unsupervised learning. In Vector Quantization (ULVQ) algorithm (Wua and Yangb, 2006), it uses the winner take all competitive learning principle in a changeable value used by several learning algorithms which effects the changing of weight values. The greater the learning rate, the more the weight values are changed.

It's usually decreased during the learning process. In competitive learning, neurons compete among themselves to be activated. While in Hebbian learning (Qian and Suen, 2000), several output neurons can be activated simultaneously. In competitive learning, only a single output neuron is active at any time. The output neuron that wins the competition is called the winner takes all neuron, only the winner neuron (or its close associates) learns. CL (Wei *et al*., 2006) includes hard learning and soft learning. Hard learning can do weight of the best winner is updated. Soft learning can do weight of winner and close associates is updated. The proposed research used Hard learning based competitive learning called as Hard Competitive Learning (HCL). Researchers give a learning (α) rate annealing schedule to improve the unsupervised learning. In this research, learning rate is 0.01. Initially, the learning rate is high. As the rate decreases, each mean gets refined. The process of decreasing the learning rate over time is called "annealing" the learning rate and it is calculated by where t is the iteration index (0 l t l NM) and N, M are number of iteration and document length, respectively (Fig. 5).

EVALUATION METRICS USED

In this study, the evaluation done by two different aspects:

- Evaluation measures: Precision, Recall and F-measure by giving user's query aspect based on information retrieval
- Evaluation of cluster quality

Evaluation measure: Number of metrics is used to measure text document representation (Manning *et al*., 2008). It is measured to know the categorization effectiveness for text document clustering. The well-known precision and recall metric are used in this study to analyse the results:

- D-set of all documents
- Q-set of documents retrieved
- R-set of relevant documents

Standard measures (Manning *et al*., 2008) in all clustering applications where the objective is to find a set of solutions. Precision is the fraction of retrieved documents that are relevant and it is equivalent to one minus fraction of retrieved documents that are false positives. Recall is the fraction of relevant documents that are retrieved and it is equivalent to one minus fraction of relevant documents that are false negatives:

$$\text{Precision} = \frac{|QR|}{|Q|} \quad (6)$$

$$\text{Recall} = \frac{|QR|}{|R|} \quad (7)$$

Precision goes up whenever the document is relevant and down every time if it is irrelevant. Both

the measures are inversely related. F-measure score is the harmonic mean of the recall and precision:

$$\text{F-measure} = \frac{2PR}{P+R} \quad (8)$$

If either p or r is small then F is small. If p and r are close then F is about the average of p and r.

Clustering quality measures: This method of clustering evaluation (Roussinov and Chen, 1999) is done by Roussinov and Chen (1999). The clustering algorithm (Michael, 2000) should be evaluated using an informative quality measure that reflects the "goodness" of the resulting clusters. To measure the quality of cluster, researchers have to calculate association by using combination of cluster (both automatic and manual) and also must calculate the number of wrong and missed associations for clustering.

Now a days datasets are grouped by an expert under some category (e.g., 20 newsgroups). This partition is called manual partition. An automatic partition is one created by the software. Inside any partition, an association is a pair of document belonging to the same cluster. Incorrect associations are those that exist in an automatic partition but do not exist in manual partition. Missed association are those that exist in a manual partition but do not exist in an automatic partition.

$$\text{Cluster error} = \frac{F}{P_t} \quad (9)$$

where, P_t is the total number of possible pairs of documents:

$$P_t = \frac{1}{2D\,(D-1)} \quad (10)$$

E represents the total number of incorrect and missed association:

$$E = E_i + E_m \quad (11)$$

This measure favours small partitions. To provide less dependence on the size of both partitions, researchers also used a normalized clustering error, expressed us:

$$\text{Normalized clustering error} = \frac{E}{A_t} \quad (12)$$

Here, A_t is the total number of all associations in both partitions without removal of duplicates. It is computed as:

$$A_t = A_m + A_\alpha \quad (13)$$

Where:

A_m = The total is number association in the manual partitions

A_α = The total number of associations in the automatic partition

Researchers have considered only associations from clusters representing three or more documents. It is easy to verify that this measure belongs to [0, 1] interval.

Cluster F-measure: External quality measures include cluster precision, cluster recall, cluster F-measure are used. It is similarly to a measure of recall and precision typically used in BoP and BoN representation. Rather than examining the number of relevant documents, researchers counted the number of correct associations:

$$\text{Cluster recall} = \frac{A_c}{A_m} \quad (14)$$

where, $A_c = A_\alpha - E_i$ represents total number of correct associations in automatic partition:

$$\text{Cluster precision} = \frac{A_c}{A_\alpha} \quad (15)$$

It is easy to check that cluster recall reflects how well the clustering technique detects association between documents. Cluster precision reflects show accurate the detected associations are. Researchers use the F-measure values to evaluate the accuracy of clustering algorithm. The F-measure is a harmonic combination of the cluster precision and cluster recall values used in information retrieval:

$$\text{Cluster F-measure} = \frac{2\times\text{Cluster precision}\times\text{Cluster recall}}{\text{Cluster precision}+\text{Cluster recall}} \quad (16)$$

EXPERIMENTS AND RESULTS

Researchers have compared the results obtained for different criterion document representation; ontology based clustering for the same dataset. The text clustering approach proposed in this study has been implemented and evaluated with extensive experimentations as follows.

Text data sets and data preparation: Researchers use two major datasets for the test including 20 Newsgroups and Reuter. A subset of 20 Newsgroups is a collection of approximately 20000 newsgroup documents, partitioned nearly evenly across 20 different newsgroups. The content of the documents are discussions made about various fields such as politics, religion, sports, science,

medicine, electronics, computers, etc. It has become a popular dataset for experiments in text applications of machine learning techniques. The original dataset contains both closely related groups and highly disjoint ones. In the test, researchers choose the following:

- Subsets of 10 relatively disjoint groups (alt.atheism, comp.graphics, comp.os.ms-windows.misc, comp.sys. ibm.pc.hardware, comp.sys.mac.hardware, comp. windows.x, misc.forsale, rec.autos, rec.motorcycles, rec.sport.baseball), it contain 50 documents each. Researchers call this dataset CVS500
- Subset of 7 relatively disjoint groups (comp.windows.x, rec.autos, sci.crypt, sci.med, talk.politics.guns, rec.sport.baseball and soc.religion. christian) each with exactly 50 documents. Researchers call this dataset NG 50
- Subset of 5 relatively disjoint groups (but comes under two main category), 50 documents each with at least 256 kB in size. Hereafter researchers call these data sets as comp.poltics, rel.sports, rec.sports (Table 1)

A subset of Reuters 21578 Reuters 21578 is currently the most widely used test collection for text categorization research. The data was originally collected and labelled by Carnegie Group, Inc. and Reuters. Because the dataset contains some noise such as repeated documents, unlabelled documents and nearly empty documents, researchers choose a subset of 10 relatively large groups (acq, coffee, crude, earn, interest, monet-fx, money-supply, ship, sugar and trade) and use two variants called here-after: RDS256 (all documents have at least 256 bytes) and RDS512 (all documents have at least 512 bytes) in the test.

Comparisons of between BoW, BoN and BoP with experimental results: In this study, researchers have been done a comparative study using Bag of Words, Bag of Phrase, Bag of Nouns with small datasets with 10 category of documents which contain 50 each) using Rapid Miner Toolkit.

The result produced by the Rapid Miner Tool kit is presented in Table 2. Researchers find that the earlier result of BoP and BoN gives the term relation (Noun, verb, etc.), reducing the dimensions and also better precision, recall than BoW representation. So, researchers have been implemented only BoP and BoN representation in the remaining part of the proposed research which uses RiTa WordNet to find the Part of Speech (POS) of term. It is used to prepare word list based on phrase.

Researchers conducted a number of experiments aimed at evaluating how the representation in the concept space can help to improve the clustering performance. According to the experimentation results, RiTa WordNet based BoN representation gives best dimensionality reduction (Fig. 6) without loss of semanticity (Table 3) than the BoP representations for various datasets. If datasets larger in size then RiTa WordNet based BoN works efficiently works than RiTa WordNet based BoP. For example, the datasets CVS 500 and RDS 1000 is larger in size than rel.sports 250. In CVS 500, RDS 1000, ontology based K-means clustering shows better precision, recall (Fig. 7), F-measure than OKM-BoP representation.

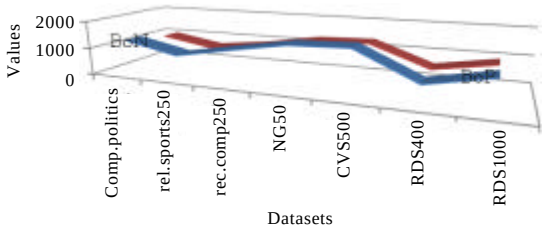

Fig. 6: Dimensions of BoP and BoN

Table 1: Summary of data sets used in experiments

Dataset	No. of doc in dataset	Classes	Dataset size
comp.politics250	250	5	687 kB
rel.sports250	250	5	528 kB
rec.comp250	250	5	473 kB
NG50	350	7	1.03 MB
CVS500	500	10	1.35 MB
RDS400	400	4	348 kB
RDS1000	1000	10	1.58 MB

Table 2: Precision and recall for BoW, BoP, BoN using rapid miner tool kit

	BoW		BoP		BoN	
Datasets	P	R	P	R	P	R
alt.atheism	1.0000	0.5800	1.0000	0.5400	1.0000	0.5000
comp.graphics	0.7083	0.6800	0.7000	0.7000	0.5556	0.6000
comp.os.ms-indows.misc	0.8333	0.7000	0.9210	0.7000	0.9444	0.6800
comp.sys.ibm.pc.hardware	0.5128	0.8000	0.5301	0.8800	0.5122	0.8400
comp.sys.mac.hardware	0.7391	0.6800	0.8995	0.6800	0.8108	0.6000
comp.windows.x	1.0000	0.3200	1.0000	0.3400	1.0000	0.3200
misc.forsale	0.7593	0.8200	0.7368	0.8400	0.3679	0.7800
rec.autos	0.4386	1.0000	0.4587	1.0000	0.5844	0.9000
rec.motorcycles	0.9444	0.6800	0.9231	0.7200	0.9688	0.6200
rec.sport.baseball	1.0000	0.7400	0.9737	0.7400	0.9429	0.6600

Table 3: Comparisons of dimensions between RiTa WordNet based BoP and RiTa WordNet based BoN representation, precision, F-measure of ontology based K-means clustering

	Dimensions		Precision		F-measure	
Datasets	BoP	BoN	OK- BoP	OK-BoN	OK-BoP	OK-BoN
comp.politics250	1238	1005	0.0040	0.022	0.0080	0.0431
rel.sports250	830	673	0.0080	0.017	0.0159	0.0334
rec.comp250	1197	906	0.0360	0.035	0.0695	0.0673
NG50	1544	1249	0.0030	0.003	0.0060	0.0060
CVS500	1614	1316	0.0100	0.021	0.0196	0.0411
RDS400	650	562	0.1110	0.003	0.0670	0.0060
RDS1000	1079	923	0.0010	0.011	0.0020	0.0206

Table 4: Precision and F-measure of ONVK-BoP and ONVK-BoN representation

	Precision		F-measure	
Datasets	ONVK-BoP	ONVK-BoN	ONVK-BoP	ONVK-BoN
comp.politics250	0.0040	0.0080	0.0040	0.0080
rel.sports250	0.0080	0.0159	0.0160	0.0315
rec.comp250	0.3600	0.5294	0.0360	0.0695
NG50	0.0030	0.0060	0.0030	0.0060
CVS500	0.0080	0.0159	0.0020	0.0040
RDS400	0.0520	0.0989	0.0020	0.0040
RDS1000	0.0010	0.0020	0.0060	0.0119

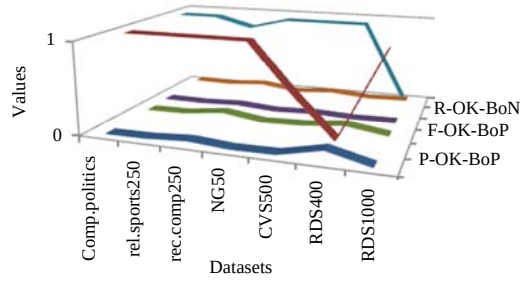

Fig. 7: P, R, F-measure of OK-BoP and OK-BoN

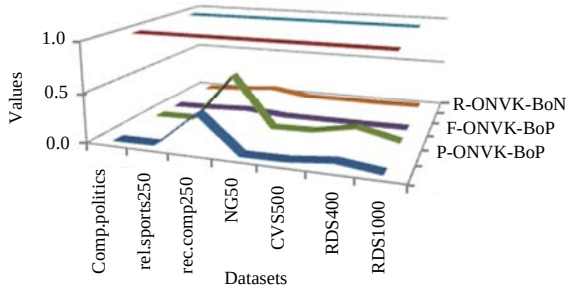

Fig. 8: P, R, F-measure of ONVK-BoP and ONVK-BoN

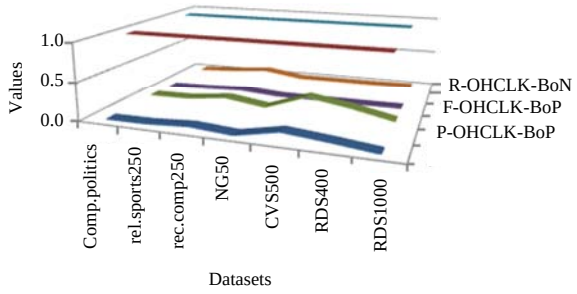

Fig. 9: P, R, F-measure of OHCLK-BoP and OHCLK-BoN

OK-BoP representation shows better precision than OKM-BoN, due to the size (528 kB) is smaller than CVS 500 and RDS 1000. Precision, recall, F-measures are calculated by giving user's query and using cosine similarity measures using based on information retrieval. ONVK-BoN (Fig. 8) is working better than ONVK-BoP representation (Table 4) in various datasets. The problem with ontology based NVK-means algorithm, it generates at least few empty cluster. OHCLK-BoN (Fig. 9) gives best

Table 5: Precision, F-measure of OHCLK-BoP and OHCLK-BoN representation

	Precision		F-measure	
Datasets	OHCLK-BoP	OHCLK-BoN	OHCLK-BoP	OHCLK-BoN
comp.politics250	0.005	0.0100	0.005	0.010
rel.sports250	0.009	0.0178	0.019	0.038
rec.comp250	0.041	0.0788	0.042	0.084
NG50	0.003	0.0060	0.003	0.006
CVS500	0.111	0.1998	0.003	0.006
RDS400	0.061	0.1150	0.003	0.006
RDS1000	0.001	0.0020	0.007	0.014

Table 6: Cluster quality: average cluster precision, average cluster recall and average cluster F-measure of ontology based HCLK-means clustering vs. ontology based K-means clustering

	Avg. cluster recall		Avg. cluster precision		Avg. cluster F-measure	
Datasets	OHCLK	OK	OHCLK	OK	OHCLK	OK
comp.politics250	0.5750	0.6117	0.7625	0.6208	0.6345	0.5795
rel.sports250	0.5150	0.4900	0.5321	0.4349	0.4405	0.3531
rec.comp250	0.5133	0.5450	0.5260	0.6244	0.4748	0.4960

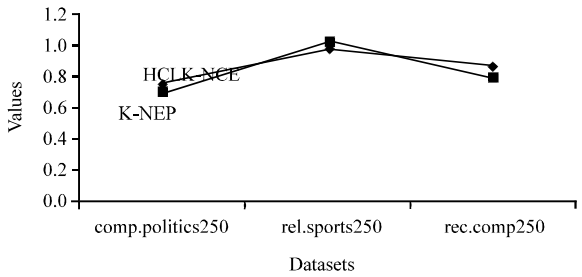

Fig. 10: Normalized cluster error of OHCLK-mean vs. OK-means

precision and F-measure than OHCLK-BoP and also better than ONVK-BoN and ONVK-BoN representation (Table 5 and 6).

Clustering results and discussion: In this study, researchers analyse the results of experiments from the different points of view. Researchers begin the experiments by first examining the effect of changing the document representation from the initial vocabulary space into the noun based concept space. A cluster must contain at least 3 documents minimum to check the cluster quality. Ontology based NVK-means (ONVK) gives better precision and recall than OK-means clustering algorithm but generate more number of empty clusters. So that, researchers cannot evaluate cluster quality of ONVK-means cluster. Researchers analyse this effect on the final clustering results of Ontology based HCLK-means (OHCK) clustering gives better average cluster precision and F-measure and normalized cluster

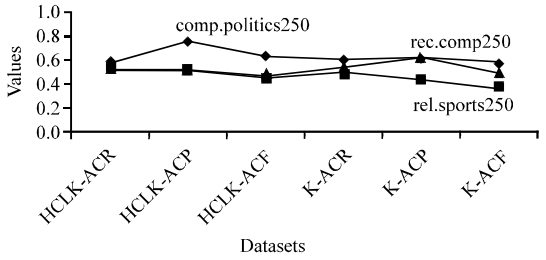

Fig. 11: Cluster quality: ACR, ACP and ACF of OHCLK-means vs. OK-means

error is also reduced than ontology based K-means clustering and avoids the empty clusters than ONVK-means clustering (Fig. 10 and 11).

CONCLUSION

The importance of document clustering emerges from the massive volumes of textual documents created. Although, numerous document clustering methods have been extensively studied in these years, there still exist several challenges for increasing the clustering quality. Entire processes, researchers begin with the process of document pre-processing and further enrich the initial representation of all documents by using RiTa WordNet in order to exploit the different Part of Speech (POS) between terms. Hard competitive learning also used for optimizes the enriched the concept space and also activate and incrementally update the cluster centroids.

The experiments reveal that the proposed algorithm has better F-measure than ontology based K-means and ontology based NVK-means based on query. The primary findings is that the approach is successful in avoiding the expansion of terms with noisy features on datasets and also for finding the important terms like unigrams or phrase, noun by using RiTa WordNet. The novel clustering OHCLK-mean performs better than OK-means.

The future research will focus that researchers should add efficient Bio-inspired algorithm to give enriched, optimized representation and syntactic attributes should be set at different weights according to their relatedness in a document. Researchers will further study whether the proposed algorithm with a syntactic analysis tool can improve the clustering results.

REFERENCES

Aliguliyev, R.M., 2009. Clustering of document collection: A weighting approach. Exp. Syst. Appli., 36: 7904-7916.

Brill, E., 1992. A Simple rule based part-of-speech Tagger. Proceedings of the 3rd Conference on Applied Natural Language Processing, March 31-April 3, 1992, Trento, Italy.

Chen, C.L., F.S.C. Tseng and T. Liang, 2010. An integration of WordNet and fuzzy association rule mining for multi-label document clustering. Data. Knowl. Eng., 69: 1208-1226.

Guerrero-Bote, V.P., C. Lopez-Pujalte, F. de Moya-Anegon and V. Herrero-Solana, 2003. Comparison of neural models for document clustering. Int. J. Approx. Reason., 34: 287-305.

Howe, D.C., 2009. Rita: Creativity support for computational literature. Proceedings of the 7th ACM Conference on Creativity and Cognition, Berkeley, CA, USA., October 27-30, 2009, ACM, New York, USA., pp: 205-210.

Kashef, R. and M.S. Kamel, 2009. Enhanced bisecting k-means clustering using intermediate cooperation. Pattern Recognition, 42: 2557-2569.

Konchady, M., 2007. Text Mining Application Programming. Charles River Publisher, Boston, Massachusetts, ISBN-13: 9781584504603.

Laia, J.Z.C. and H. Tsung-Jen, 2010. Fast global *K*-Means clustering cluster membership and inequality. Pattern Recognition, 43: 1954-1963.

Manning, C.D., P. Raghavan and H. Schutze, 2008. An Introduction to Information Retrieval. Cambridge University Press, Cambridge, UK., ISBN-13: 9780521865715, pp: 482.

Michael, K. Ng, 2000. A note on constrained k-means algorithms. Pattern Recognition, 33: 515-519.

Pessiot, J.F., Y.M. Kim, M.R. Amini and P. Gallinari, 2010. Improving document clustering in a learned concept space. Inform. Proc. Manage., 46: 180-192.

Pullwitt, D., 2002. Integrating contexual information to enhance SOM-based text document clustering. Neural Networks, 15: 1099-1106.

Qian, Y. and C.Y. Suen, 2000. Clustering combination method. Proceedings of the International Conference in Pattern Recognition. Volume 2, September 3-7, 2000, Barcelona, pp: 732-735.

Roussinov, D.G. and H. Chen, 1999. Document clustering for electronic meetings: An experimental comparison of two techniques. Deci. Sup. Syst., 27: 67-80.

Shafiei, M., S. Wang, R. Zhang, E. Milios and B. Tang *et al.*, 2007. Document Representation and dimension reduction for text clustering. Proceedings of the 23rd International Conference on Data Engineering Workshop, April 17-20, 2007, Istanbul, pp: 770-779.

Wei, C.P., C.S. Yang, H.W. Hsiao and T.H. Cheng, 2006. Combining preference- and content-based approaches for improving document clustering effectiveness. Inform. Proces. Manage., 42: 350-372.

Wua, K.L. and M.S. Yangb, 2006. Alternative learning vector quantization. Pattern Recognition, 39: 351-362.

Zhang, W., T. Yoshida, X. Tang and Q. Wang, 2010. Text clustering using frequent itemsets. Knowledge-Based Syst., 23: 379-388.

Zheng, H.T., C. Borchert and H.G. Kim, 2010. GOClonto: An ontological clustering approach for conceptualizing PubMed abstracts. J. Biomed. Inform., 43: 31-40.

Zhong, S., 2005. Efficient online spherical k-means clustering. Proceedings of the IEEE International Joint Conference on Neural Networks, Volume 5, July 31-August 4, 2005, Montreal, QC, Canada, pp: 3180-3185.