DESIGN OF FUZZY BASED CLUSTERING ALGORITHM
CHAPTER 5

5. DESIGN OF FUZZY BASED CLUSTERING ALGORITHM

5.1 INTRODUCTION

As considered several approaches in clustering of data based on a certain pattern, the similarities play a vital role in clustering sentences on the prediction in order to produce an efficient result comparing to the previous approaches. Hence, a fuzzy logic based approach has been proposed for finding the similarities to form a cluster, based on the relational prototypes. A semantic clustering and fuzzy based categorical text clustering approach is practiced to bring more accuracy in mining process. The algorithm indentifies the semantically related sentences and avoids duplication on the given data set. The information retrieval based on the proposed algorithm will maximize the accuracy compared to the earlier ones.

5.2 OVERVIEW OF FUZZY CATEGORICAL CLUSTERING

The fuzzy clustering algorithm that can in principle be applied to any relational clustering problem. But eventually, it depends on the excellence of the dataset and in the case of sentence clustering, this efficiency in performance may be achieved through the sentence similarity measures which may in turn be based on improved word sense disambiguation. This research work further improves the Fuzzy categorical text clustering to reduce the time consumption which was considered as the major limitation on the existing approaches. In text processing, the major part is sentence clustering since sentence clustering is nothing but grouping of sentences which are similar meanings into clusters. Then, the task is performed by applying standard clustering algorithms to group sentences into clusters. Various traditional methods such as STIRR, ROCK and k-Modes are followed in practice that represents the sentences as vectors in term space and applies best clustering algorithm to achieve the result accuracy. A novel sentence clustering scheme has been presented based on fuzzy logic on sentences over the term clusters. The major challenge in implementing the sentence clustering approach is language variability where the same meaning can
be phrased in various ways. The shorter the sentences are the less effective becomes exact matching of their terms. Various traditional methods in text processing are mainly focused on reducing dimensionality, removing irrelevant data, increasing learning accuracy and improving result comprehensibility. The embedded methods involves in feature selection which to be a part of training process that are usually meant for learning algorithms. Traditional machine learning algorithms such as decision trees and artificial neural networks are discussed which are all depend on the embedded approaches.

**Decision tree**

Decision trees are powerful and popular tools for classification and prediction. The attractiveness of decision trees is due to the fact that, in contrast to neural networks, decision trees represent rules. Rules can readily be expressed so that humans can understand them or even directly used in a database access language so that records falling into a particular category may be retrieved. Decision tree learning is one of the most successful techniques for supervised classification learning. The goal is to create a model that predicts the value of a target variable based on several input variables. Assume that all of the features have finite discrete domains and there is a single target feature called the classification. Each element of the domain of the classification is called a class. A decision tree or a classification tree is a tree in which each internal (non-leaf) node is labelled with an input feature. The arcs coming from a node labelled with a feature are labelled with each of the possible values of the feature. Each leaf of the tree is labelled with a class or a probability distribution over the classes [Rokach, Lior and Maimon, O. 2008]. In data mining, decision trees can be described also as the combination of mathematical and computational techniques to aid the description, categorisation and generalisation of a given set of data. In some applications, the accuracy of a classification or prediction is the only thing that matters. In some situations, the ability to explain the reason for a decision is crucial.

**Artificial Neural Network**

Neural networks are of particular interest because they offer a means of efficiently modeling large and complex problems. Neural networks may be used in classification problems or for regressions. An artificial neural network can be defined as an information processing system consisting of many processing elements which
are joined together in a structure inspired by the cerebral cortex of the brain. The processing elements considered in the definition of ANN are usually organized in a sequence of layers, with full connections between layers. Typically, there are three (or more) layers: an input layer where data are presented to the network through an input buffer, an output layer with a buffer that holds the output response to a given input and one or more intermediate or hidden layers. (Figure 5.1)

The operation of an artificial neural network involves two processes: learning and recall. Learning is the process of updating the connection weights in response to external stimuli presented at the input buffer. The network “learns” in accordance with a learning rule governing the adjustment of connection weights in response to learning examples applied at the input and output buffers. Recall is the process of accepting an input and producing a response determined by the geometry and synaptic weights of the network.

![Figure 5.1 Artificial Neural Networks](image)

The initial approach in this method is feature reduction. After giving input dataset, it undergoes several pre-processing based on that a result will be generated. The result is passed to the word net that avoids the words which are all not meaningful and those unwanted words. After that, the resultant data is processed on the fuzzy logic to get the categorical text clustering.
5.3 COMPUTATIONAL COMPLEXITY

Generally, the text clustering mainly focused on reducing dimensionality, removing irrelevant data, increasing learning accuracy and improving result comprehensibility. The solution produced by the existing algorithms like STIRR, ROCK & k-modes are not effective on above mentioned problems. The high dimensional vector space in which each aspect corresponds to a unique keyword. Along with this, the problem of extracting representative sentences from text is also not effective. Traditional machine learning algorithms like decision trees or artificial neural networks are examples of embedded approaches. The wrapper methods use the predictive accuracy of a pre-determined learning algorithm to determine the goodness of the selected subsets, the accuracy of the learning algorithms is usually high. However, the generality of the selected features is limited and the computational complexity is large and the filter methods are independent of learning.

The computational complexity of wrapper methods is low, but the accuracy of the learning algorithms is not guaranteed. The hybrid methods are a combination of filter and wrapper methods by using a filter method to reduce search space that will be considered by the subsequent wrapper. They mainly focus on combining filter and wrapper methods to achieve the best possible performance with a particular learning algorithm with similar time complexity of the filter methods. The wrapper methods are computationally expensive and tend to over fit on small training sets. The filter methods, in addition to their generality, are usually a good choice when the number of features is very large. With respect to the filter feature selection methods, the application of cluster analysis has been demonstrated to be more effective than traditional feature selection algorithms.

Feature subset selection can be viewed as the process of identifying and removing as many irrelevant and redundant features as possible. This is because irrelevant features do not contribute to the predictive accuracy and redundant features do not redound to getting a better predictor for that they provide mostly information which is already present in other feature(s). Of the many feature subset selection algorithms, few algorithms effectively eliminate irrelevant features but fail to handle redundant features yet some others can eliminate the irrelevant while taking
care of the redundant features. Traditionally, feature subset selection research has focused on searching for relevant features.

5.4 DESIGN OF ATTRIBUTES

The number of initial clusters must be specified as input to the algorithm. If this number is too high, then duplicate clusters (i.e. clusters with identical membership values across all objects) are found. While it might appear at first sight that duplicate clusters can simply be removed after the algorithm has converged and membership subsequently renormalized to sum to one, this is not possible because of the coupling between membership values and Page Rank values. That is, it cannot be assumed that the current Page Rank values are correct under a renormalization of membership values. The solution is to perform a check for duplicate clusters at the completion of each maximization step. If duplicate clusters are found, membership values are renormalized and the algorithm is allowed to proceed until a stage at which convergence has been achieved and no duplicate clusters exist.

5.4.1 THRESHOLDING OF SIMILARITY VALUES

Depending on the domain, the graph representing the relation between objects may be heavily or sparsely connected. In the case of sentence clustering, it is found that many of the similarities $S_{ij}$ between sentences are very small and are likely to be spurious, arising from incidental similarities between words in sentences which are in fact not semantically related. In practice, it is also found that the clustering performance of the algorithm can be improved by thresholding. These similarity values are such that all values below the threshold are converted to zero. All sentence clustering results reported in this research works are based on thresholding similarity values such that 50 percent of the values in the affinity matrix are zero.

5.4.2 HARD CLUSTERING

Hard clustering methods are based on classical set theory and require that an object either does or does not belong to a cluster. Hard clustering means partitioning the data into a specified number of mutually exclusive subsets.
The algorithm outputs cluster membership values $k, i$, which represent the degree of membership of object $i$ to cluster $k$. If hard clustering is required, this can be trivially achieved by assigning a sentence to the cluster $m$ for which membership is highest.

5.5 SOLUTION STRATEGIES

The aim of a text clustering scheme is to minimizing intra-cluster distances between text and maximizing inter-cluster distances (using an appropriate distance measure between documents). A distance measure (or, dually, similarity measure) thus lies at the heart of text clustering. The large variety of documents makes it almost impossible to create a general algorithm which can work best in case of all kinds of datasets. The major use of document clustering is to give users an overview of the contents of a document collection and reduce search space. If a collection is well clustered, search only the clusters that will contain relevant text. Efficiency and effectiveness can be improved by searching through smaller collection itself.

Text categorization is the task of assigning a Boolean value to each pair $\langle d_j, c_i \rangle \in D \times C$, where $D$ is a set of documents and $C = \{c_1, \ldots, c_k\}$ is a set of predefined categories. A value of T assigned to $\langle d_j, c_i \rangle$ indicates a decision to file $d_j$ under $c_i$, while a value of F indicates a decision not to file $d_j$ under $c_i$. In general, the task is to approximate the unknown target function: $f^*: D \times C \rightarrow \{\text{true}, \text{false}\}$ that describes how documents ought to be classified by means of a function $f: D \times C \rightarrow \{\text{true}, \text{false}\}$ called the classifier such that $f^*$ and $f$ coincide as much as possible.

Different constraints may be enforced on the Text Clustering (TC) task, depending on the application. TC may be either a single-labeled (i.e. exactly one $c_i \in C$ must be assigned to each $d_j \in D$) or a multi-labeled (i.e. any number $0 \leq n_j \leq |C|$ of categories may be assigned to a document $d_j \in D$). A special case of single-labeled TC is binary TC, in which, given category $c_i$, each $d_j \in D$ must be assigned either to $c_i$ or to its complement $\overline{c_i}$. A classifier for $c_i$ is then a function $f_i^*: D \rightarrow \{\text{true}, \text{false}\}$ that approximates the unknown target function $f_i: D \rightarrow \{\text{true}, \text{false}\}$. The problem of multi-label TC under $C = \{c_1, \ldots, \ldots, c_k\}$ is usually tackled.
as $|C|$ independent binary classification problems under $\{ \overline{c}, c_i \}$, for $i = 1, \ldots, |C|$. In this case, a classifier for $C$ is thus actually composed of $|C|$ binary classifiers.

5.5.1 CLUSTERING FAMOUS QUOTATION

The algorithm is applied to clustering famous quotations and they provide a rich and challenging context for evaluating sentence clustering because they often contain a lot of semantic information (i.e. wisdom packed into a small message) and are often couched in a poetic use of language. An external clustering evaluation criterion has been applied (which requires that the true groupings is known) and then compiled a database of famous quotations from five different classes. The quotations are taken from an extract. Although there is some degree of word overlap between quotations, this is not sufficient to allow adequate measure of similarity by using a conventional bag of words approach.

The quality of data affects the data mining results. In order to improve the quality of data and consequently of the mining results, raw data is pre-processed so as to improve the efficiency and ease of mining process. In the proposed system, pre-processing for dataset is done to remove the stop words and stem words which are considered as less important and to improve quality and efficiency of data. Many of the most frequently used words in English are useless in Information Retrieval (IR) and text mining. Stop-words, which are language-specific functional words, are frequent words that carry no information (i.e. pronouns, prepositions, conjunctions). Examples of such words include 'the', 'of',' and', 'to', etc. These stop words are get stored in the database. Dataset (famous quotations) is loaded into another database. The stop words in data set (famous quotation) are removed by comparing with the stop word database. The number of initial clusters must be specified as input to the algorithm. If this number is too high, then duplicate clusters (i.e. clusters with identical membership values across all objects) are found.

Sentence clustering plays an important role in many text processing activities. For example, various authors have argued that incorporating sentence clustering into extractive multi-document summarization helps avoid problems of content overlap, leading to better coverage. However, sentence clustering is used within more general text mining tasks. For example, consider web mining, where the specific objective
might be to discover relevant novel information from a set of documents initially retrieved in response to some query. By clustering the sentences of those documents it would intuitively expected that at least one of the clusters to be closely related to the concepts described by the query terms; however, other clusters may contain information pertaining to the query in some way hitherto unknown and in such a case it would have successfully mined new information.

The most quotations will contain interrelated topics or themes and many sentences will be related to some degree to a number of these. The work described in this research work is motivated by the belief that successfully being able to capture such fuzzy relationships will lead to an increase in the breadth and scope of problems to which sentence clustering can be applied. However, clustering text at the sentence level poses specific challenges not present when clustering larger segments of text such as documents.

The famous quotations data sets have been constructed in order to evaluate the performance of the algorithm by using standard external cluster quality criteria. To demonstrate how the algorithm may be of more general use in activities related to text mining, the algorithm has been applied to clustering sentences from a recent news article.

5.5.2 MEASURING SENTENCE SIMILARITY

Document comparison process generates frequent term sets for the given document with minimum support from 5 to 95 after subjected to the document pre-processing task (stopword removal and stemming process). Then, it is compared with the trained document available in the database one by one and finds out the matching percentage. Binary searching technique is used to search a term from the trained document database during matching process because of its competency/efficiency. The procedure accepts the document name to perform stopword removal, stemming and frequent term set generation pre-processing steps. Initially, a minimum support of 5% is considered to generate frequent term set. Then this frequent term sets are compared with every trained document in the database to determine the matching percentage after that the minimum support is increased by 5% and then performs the matching percentage once again with the entire trained document to evaluate the
matching percentage. This process is repeated until the matching percentage is reached to 95% or maximum (all) document matching becomes 100%.

To calculate similarity values $S_{ij}$ for the affinity matrix, a modified version of the measure has been proposed. This approach is similar to that used to calculate document similarity in the IR literature; however, rather than using a common vector space representation for all sentences, the two sentences being compared are represented in a reduced vector space of dimension $n$, where $n$ is the number of distinct nonstop words appearing in the two sentences. Semantic vectors, $V_1$ and $V_2$, representing sentences $S_1$ and $S_2$ in the reduced vector space are first constructed. The elements of $V_i$ are determined as follows: Let $V_{ij}$ be the $j$th element of $V_i$ and let $w_j$ be the word corresponding to dimension $j$ in the reduced vector space. Once $V_1$ and $V_2$ have been determined, the semantic similarity between $S_1$ and $S_2$ is defined by using a standard measure of similarity. The sentence similarity measure relies on a word-to-word semantic similarity measure. Many such measures have been proposed and can broadly be categorized as either corpus-based, in which case the similarity is calculated based on distributional information derived from large corpora and knowledge-based, in which similarity is based on semantic relations expressed in external resources such as dictionaries or thesauri.

$$sim(v_1, v_2) = \frac{1}{IC(v_1) + IC(v_2) - 2 \cdot IC(LCS(v_1, v_2))}$$

Applying this distance formula to a word sense disambiguation task, an improvement where multiple sense words have been disambiguated by finding the combination of senses from a set of contiguous terms which minimizes total pairwise distance between senses. Hence, it is found that the performance is robust under a number of perturbations; however, depth factor scaling and restricting the type of link to a strictly hierarchical relation do noticeably impair performance.

Similarity calculation is mainly based on number of terms which is common between two sentences by number of words present in both sentences. Based on sentence similarity, sentences with highest Page Rank value are taken through Page Rank algorithm. Page Rank algorithm provides the importance of sentence i.e. how many times the sentence appears in the document and then fuzzy clustering algorithm
is applied. Mixing coefficients are initialized such that priors for all clusters are equal. In expectation step, Page Rank value for each object in each cluster is calculated. Page Rank algorithm provides the importance of sentence i.e. how many times the sentence appears in the document. Maximization step involves only the single step of updating the mixing coefficients based on membership values calculated in the Expectation Step.

The ability to accurately judge the similarity between natural language sentences is critical to the performance of several applications such as text mining, question answering and text summarization. Given two sentences, an effective similarity measure should be able to determine whether the sentences are semantically equivalent or not, taking into account the variability of natural language expression. Similarity between two sentences is provided by Text rank measure. Assume that cluster membership values are initialized randomly and normalized such that cluster membership for an object sums to unity over all clusters. Mixing coefficients are initialized such that priors for all clusters are equal.

The dataset used for this research work is famous quotation dataset where a large number of documents are available for usage and they are analyzed offline. There are certain advantages in the work of semantic association discovery by combining a taxonomy structure with corpus statistics. The incorporation of a manually built pseudo knowledge base (e.g. thesaurus or taxonomy) may complement the statistical approach where “true” understanding of the text is unobtainable. By doing this, the statistics model can take advantage of a conceptual space structured by a hand-crafted taxonomy, while providing computational evidence from manoeuvring in the conceptual space via distributional analysis of data. In other words, calculating the semantic association can be transformed to the estimation of the conceptual similarity (or distance) between nodes (words or concepts) in the conceptual space generated by the taxonomy. Ideally, this kind of knowledge base should be reasonably broad-coverage, well-structured and easily manipulated in order to derive desired associative or similarity information.
5.6 FUZZY BASED ARCHITECTURAL DESIGN

Based on the above observation, Fuzzy based algorithm has been proposed in which the result belongs to a single cluster. A semantic clustering and fuzzy based pruning approach is practiced to bring more accuracy in mining process. Generally, fuzzy clustering based on the prototypes or mixtures of Gaussians which does not support sentence clustering. The algorithm indentifies the semantically related sentences and avoids duplication on the given data set. The information retrieval based on the keyword in which filtering is processed on the benchmark dataset. Fuzzy sets are closely related to the definition of similarities because of their capacity to represent subjective information, resulting from real world complexity and gray areas of interpretation and because of the graduality inherent in their definition, in agreement with the natural behavior of intuitive similarities. An overview of similarities has been proposed in the framework of fuzzy logic, similarity measures enabling the user to preserve the flexibility and graduality human beings have in mind when they deal with similarities and use expressions such as "very similar", "rather similar", "more similar than", etc.,

Fuzzy clustering is important in domains such as sentence clustering, since a sentence is related to more than one theme or topic present within a document or set of documents. In the proposed system, Fuzzy clustering algorithm operates on Expectation-Maximization framework in which the cluster membership probabilities for sentence in each cluster are identified. Results obtained while applying the algorithm to sentence clustering tasks demonstrate that the algorithm is capable of identifying overlapping clusters of semantically related sentences and its performance improvement can be proved by comparing with k-medoid. Performance measures such as Purity, Entropy, Partition-Entropy and V-Measure are used to prove the performance improvement of document clustering and its application in document summarization.

The research work demonstrates to stress on the fact that many of the measures, methods and properties which are pointed out can be used in a general environment, not necessarily involving a fuzzy set based representation, the fuzziness and graduality appearing in the only similarities themselves. There exist various types of similarity measures: for binary data, for fuzzy data, for numerical data, for
structured data, etc., This work focuses on classical definitions related to binary data
and their extensions to fuzzy data in a general formalization incorporating most of the
well known measures, classifying them and proposing new ones, to help the user to
choose one of them according to the problem.

Fuzzy uses weighting schemes for the process of information retrieval in
which it also assess the importance of whole attributes and individual values in the
dataset. The work is intended immediate retrieval of response based on the input
query. Based on the query, the clusters are done which related to the concepts based
on the given queries by the user. The page rank algorithm has been used which was
developed by Diane Kelly [2009].

\[ xy(v_i) = (1 - d) \sum_{j \in \text{cent}(v_j)} \frac{1}{\text{out}(v_j)} xy(v_j) \]

where \( v_i \) points in and \( v_j \) points out on the set of vertices. The similarity between the \( v_j \)
and \( v_i \) based on the similarity as store these are stored on the matrix form such as
\( W = (w_{ij}) \) that refers to the affinity matrix. This can expressed on the below equations
and the following steps.

\[ xy(v_i) = (1 - d) + d \sum_{j=1}^{N} \frac{xy(v_j)}{\sum_{k=1}^{N} w_{jk}} \]

1. Compute the similarity between all pairs of clusters i.e. calculates a
similarity matrix whose \( ij^{th} \) entry gives the similarity between the \( i^{th} \) and \( j^{th} \)
clusters.

2. Merge the most similar (closest) two clusters.

3. Update the similarity matrix to reflect the pairwise similarity between the
new cluster and the original clusters.

4. Repeat steps 2 and 3 until only a single cluster remains.
The overall Fuzzy based categorical text clustering steps is given in Figure 5.2.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Fragment each sentence into single words and those words are passed into the word net</td>
</tr>
<tr>
<td>2.</td>
<td>Filter conjunctions and keywords</td>
</tr>
<tr>
<td>3.</td>
<td>Retrieval decision is made by comparing the terms of query with index terms</td>
</tr>
<tr>
<td>4.</td>
<td>Validate the duplication in the word net and determine the frequency of occurrence</td>
</tr>
<tr>
<td>5.</td>
<td>Form semantic clustering based on Fuzzy based pruning approach</td>
</tr>
<tr>
<td>a.</td>
<td>The similarity matrix can be formed using ( \text{sim} = {\text{sim}_{xy}} ) where ( x ) and ( y ) is the similarity between the objects.</td>
</tr>
<tr>
<td>6.</td>
<td>The weight between the matrices is ( w_{ij}^c ) where ( c ) is the cluster</td>
</tr>
<tr>
<td>7.</td>
<td>Similarity matrix is calculated using ( \text{Sim}_{x&amp;y} (w1, w2) = 1/(\text{IC} (w1) +\text{IC} (w2) -2*\text{IC} (\text{DCS} (w1, w2))) ) where ( w ) are the words from the information content, ( \text{IC} ) are the information content and ( \text{DCS} ) is deepest common similarity.</td>
</tr>
<tr>
<td>8.</td>
<td>( \text{IC} (w) ) is calculated by ( \text{IC} (w) = -\log P (w) ) that is probability of the word ( w ) appear in the IC information content.</td>
</tr>
</tbody>
</table>

**Figure 5.2 Fuzzy based categorical text clustering algorithm**

The results of the experiment confirm that the information content approach proposed provides a significant improvement over the traditional edge counting method. It also shows that the proposed combined fuzzy approach outperforms the information content approach. One should recognize that even a small percentage improvement over the existing approaches is of significance since the results are nearing the observed upper bound.

**5.7 ARCHITECTURAL DESIGN**

As a document set may contain several thousands of words, it results in a very high impracticable dimensionality. To reduce the document space dimensionality, word reduction methods are applied in the pre-processing phase. The most common method to reduce the number of different words is to eliminate the words with low information value. The stemming algorithm is used in text clustering to remove suffixes from the words in order to determine the common root of the words.
Figure 5.3 Architecture of fuzzy based categorical text clustering system
Another way to reduce the document space dimensionality is based on the statistical properties of the word. The infrequent and frequent words are filtered out from the original text. The reduction is based on the assumption that, in the words with low frequency, is not a characteristic word for the document set. The weight of a word is measured by its frequency count with respect to minimum support threshold value.

The steps involved in the architecture shown in Figure 5.3 are discussed below:

**Collection of Data:** The famous quotations data set was constructed in order to evaluate performance of the algorithm using standard external cluster quality criteria. To demonstrate how the algorithm may be of more general use in activities related to text mining, the algorithm has been applied to clustering sentences from a recent news article includes the processes like crawling, indexing, filtering etc., which are used to collect the documents that need to be clustered, index them to store and retrieve in a better way and filter them to remove the extra data, for example, stop words. The number of documents on the internet is continuously increasing due to large amount of online sources available and it is very difficult for the users to go through all the sources and find the relevant information from the collection.

The sentences of those documents, at least one of the clusters to be closely related to the concepts described by the query terms; however, other clusters may contain information pertaining to the query in some way hitherto unknown to us and in such a case new information is mined successfully. Irrespective of the specific task (e.g. summarization, text mining, etc.), most documents will contain interrelated topics or themes and many sentences will be related to some degree to a number of these. This work concentrates in determining the global importance of a sentence; i.e. how important a sentence is in the context of the paper as a whole, irrespective of its membership to individual clusters.


**Pre-processing:** Raw data is highly concerned with noise, missing values and inconsistency and the quality of data affects the data mining results. In order to improve the quality of data and consequently of the mining results, raw data is pre-processed so as to improve the efficiency and ease of mining process. In the proposed
system, pre-processing for dataset is done to remove the stop words and stem words which are considered as less important and to improve quality and efficiency of data. Many of the most frequently used words in English are useless in Information Retrieval and text mining. These words are called ‘Stopwords’. It consists of steps that take as input a plain text document and output a set of tokens (which can be single terms or n-grams) to be included in the vector model. Filtering is the process of removing special characters and punctuation that are not thought to hold any discriminative power under the vector model. This is more critical in the case of formatted documents, such as web pages where formatting tags can either be discarded or identified and their constituent terms attributed different weights. Term Frequency Weighting methods are applicable only when the selected terms of the documents are known in advance. In new document environment, frequent term set generation is the most applicable method to find frequent terms as well as for dimension reduction. Frequent terms are determined by scanning the document and collecting those terms that satisfy the minimum support threshold value.

**Tokenization:** Tokenization splits sentences into individual tokens, typically words. More sophisticated methods, drawn from the field of NLP, parse the grammatical structure of the text to pick significant terms or chunks, such as noun phrases. Stemming is the process of reducing words to their base form or stem. For example, the words “connected”, “connection”, “connections” are all reduced to the stem “connect”. Porter's algorithm is the de facto standard stemming algorithm.

**Stop word removal:** A stop word is defined as a term, which is not thought to convey any meaning as a dimension in the vector space (i.e. without context). A typical method to remove stop words is to compare each term with a compilation of known stop words. Another approach is to first apply a parts-of-speech tagger and then reject all tokens that are not nouns, verbs and adjectives.

In computer search engines, a stop word is a commonly used word (such as “the”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query. When building the index, most engines are programmed to remove certain words from any index entry. The list of words that are not to be added is called a stop list. Stop words are deemed irrelevant for searching purposes because they occur frequently in the
language for which the indexing engine has been tuned. In order to save both space and time, these words are dropped at indexing time and then ignored at search time. Some search engines allow one to include a stop word in a search by putting an inclusion (plus sign) before each stop word in the query. Sometimes, some extremely common words which would appear to be of little value in helping select documents matching a user need are excluded from the vocabulary entirely (stopwords). The general strategy for determining a stop list is to sort the terms by collection frequency (the total number of times each term appears in the document collection) and then to take the most frequent terms, often hand-filtered for their semantic content relative to the domain of the documents being indexed, as a stop list, the members of which are then discarded during indexing. Stop words are removed from the index and the query. Words might carry little meaning from a frequency (or information theoretic) point of view and alternatively from a linguistic point of view. Words that occur in many of the documents in the collection carry little meaning from a frequency point of view.

**Stemming**: One challenge emerging, when terms are defined as single words, is that the feature space becomes very highly dimensional. In addition, words which are in the same context, such as biology and biologist are defined as different terms. So, in order to define words that are in the same context with the same term and consequently to reduce dimensionality, the terms are defined as stemmed words. A stemmer applies morphological ‘rules of the thumb’ to normalize words. Stemming tends to help as many queries as it hurts. Sometimes stemming algorithms may conflate two words with very different meanings to the same stem, for instance the words “skies” and “ski” will both be reduced to “ski”. In such cases users might not understand why a certain document is retrieved and may begin to question the integrity of the system in general. Still, stemmers are used often in many research systems like Smart, Okapi and Twenty-One. The In Query System uses a stemming technique called K-stem that combines dictionary lookup and stemming rules.

In most cases, morphological variants of words have similar semantic interpretations and are considered as equivalent for the purpose of IR applications. For this reason, a number of so-called stemming algorithms and stemmers, have been developed, which attempt to reduce a word to its stem or root form. Thus, the key
terms of a query or document are represented by stems rather than by the original words. This not only means that different variants of a term can be conflated to a single representative form and it also reduces the dictionary size, that is, the number of distinct terms needed for representing a set of documents. A smaller dictionary size results in a saving of storage space and processing time.

**Keyword stemming:** Keyword stemming is a useful tool for web pages and search engine optimization. The process of keyword stemming involves taking a basic but popular keyword pertaining to a particular website and adding a prefix, suffix and pluralization to make the keyword into a new word. This particular process allows a website to expand upon the number of variable options, which can help a website get more traffic. Words that are a product of keyword stemming can expand in either direction and even add words to the phrase, making the possibilities limitless.

**Sentence Similarity calculation:** The ability to accurately judge the similarity between natural language sentences is critical to the performance of several applications such as text mining, question answering and text summarization. Given two sentences, an effective similarity measure should be able to determine whether the sentences are semantically equivalent or not, taking into account the variability of natural language expression. Similarity between two sentences is provided by Text Rank measure. A different approach proposes to extract the dissimilarity relation directly from the data by guiding the extraction process itself with as little supervision as possible.

\[
\text{Similarity} (S_i, S_j) = \{ W_k | W_k \in S_i, W_k \in S_j \}/\log(|S_i|) + \log (|S_j|)
\]

where \(W_k\) denotes number of terms common between two sentences \((S_i, S_j)\)

\[
\log(|S_i|) \text{ denotes number of words in sentence } i \text{ and }
\]

\[
\log (|S_j|) \text{ denotes number of words in sentence } j
\]

**Page Value:** By means of Page Rank algorithm, Page Rank value for each sentence is calculated. Calculating the importance of a sentence is that sentences which are similar to a large number of other important sentences are central. Thus, by ranking sentences according to their centrality, the top ranking sentences can then be extracted and provided as input to our proposed algorithm.
Pruning: Pruning removes words that appear with very low frequency throughout the corpus. The underlying assumption is that these words, even if they had any discriminating power, would form too small clusters to be useful. A pre-specified threshold is typically used, e.g. a small fraction of the number of words in the corpus. Sometimes words which occur too frequently (e.g. in 40% or more of the documents) are also removed.

Post processing: Post processing includes the major applications, in which the document clustering is used, e.g. the application that uses the results of clustering for recommending news articles to the users.

5.8 DISCUSSION

The problem of text clustering is generally defined as follows: given a set of document clustering. An automatically derived number of clusters, such that the documents assigned to each cluster are more similar to each other than the documents assigned to different clusters. Texts are represented by using the vector space model that treats a document as a bag of words. A major characteristic of document clustering algorithms is the high dimensionality of the feature space, which imposes a big challenge to the performance of clustering algorithms. They could not work efficiently in high dimensional feature spaces due to the inherent sparseness of the data. The next challenge is that not all features are important for document clustering, some of the features may be redundant or irrelevant and some may even misguide the clustering result, especially there are more irrelevant more features than relevant ones.

The wrapper methods use the predictive accuracy of a predetermined learning algorithm to determine the goodness of the selected subsets, the accuracy of the learning algorithms is usually high. However, the generality of the selected features is limited and the computational complexity is large and the filter methods are independent of learning. Fuzzy based algorithm has been proposed in which the result belongs to a single cluster and a semantic clustering and fuzzy based pruning approach is practiced to bring more accuracy in mining process. Fuzzy uses weighting schemes for the process of information retrieval in which it also assess the importance of whole attributes and individual values in the dataset. The works is intended immediate retrieval of response based on the input query. Based on the query, the
clusters are done which are related to the concepts based on the given queries by the user.

The famous quotations data set was constructed in order to evaluate performance of the algorithm by using standard external cluster quality criteria. The overall algorithm is presented to prove the concepts used. To demonstrate how the algorithm may be of more general use in activities related to text mining, the algorithm has been applied to clustering sentences from a recent news article.