Ontology Based Fuzzy Document Clustering Scheme

Thangamani.M
Department of Computer Technology
School of Computer Technology and Applications
Kongu Engineering College, Perundurai-638 052, Erode(District), Tamil Nadu, India
Tel: 94-88-152-464 E-mail: vetha_narayana@yahoo.co.in

Dr. Thangaraj.P
School of Computer Technology and Applications
Kongu Engineering College, Perundurai-638 052, Erode(District), Tamil Nadu, India
Tel: 91-04294-220-562 E-mail: ctptr@yahoo.co.in

Abstract

Document clustering is the technique used to group up the document with the reference to the similarity. It is widely used in web mining and digital library environment. Documents are represented in vector space model. Each document is a vector in the word space and each element of the vector indicates the frequency of the corresponding word in the document. Documents are presented as high dimensional data elements. It is a very complex task to cluster documents using K-means clustering algorithm. The sub space clustering schemes can be adopted to cluster documents. The document clustering uses the term weights from the similarity measure. The sub space model uses the relevant attributes for the similarity estimation. The fuzzy logic is used to cluster the documents. The fuzzy document clustering scheme is enhanced with semantic analysis mechanism. Semantic analysis is carried out with the support of the ontology. The ontology is used to maintain term relationships. Term relationships are represented using the synonym, meronym and hypernym factors. Ontology is manually collected by the users. Domain based ontology is used for the document clustering process. The system uses the data mining domain based ontology for the semantic analysis. Semantic weights are used in the similarity measure. Fuzzy based text document clustering scheme uses the stop word filters and stemming process under the document preprocess. Term clustering and semantic clustering operations are performed in the system.

Keywords: Fuzzy clustering, Fuzzy, Document clustering, Data mining, Fuzzy ontology, Semantic web

1. Introduction

Document clustering has been studied intensively because of its wide applicability in areas such as web mining and information retrieval. In document clustering, unlabeled documents are typically represented in vector space model (VSM), where each document is a vector in the word space and each element of the vector indicates the frequency of the corresponding word (also called term or feature) in the document. Generally, the data are of very high dimensional and sparse, which poses a big challenge to conventional clustering algorithms such as k-means (S.B.Kotsiantis and P.E.Pintelas. 2004).In high dimensional data, clusters often exist in subspaces rather than in the entire space (L.Jing, M.K.Ng, and J.Z.Huang. 2007). For example, in document clustering, clusters of documents of different topics are categorized by different subsets of keywords. Moreover, the keywords for one cluster may not occur in the documents of other clusters. One solution to this problem is text subspace clustering (L.Jing, M.K.Ng, J.Xu, and J.Z.Huang. 2005), which aims to discovering the document clusters in different subspaces of the original word space. In the past few years, soft subspace clustering algorithms have been developed and successfully applied to clustering large document collections. Examples includes LAC (C.Domeniconi, D.Gunopulos, and S.Ma. 2006), FWKM (L.Jing, M.K.Ng, J.Xu, and J.Z.Huang. 2005), [5] and EWKM (L.Jing, M.K.Ng, and J.Z.Huang. 2007) etc. In these algorithms, each term is assigned with a desired set of weighting values to distinguish its different contributions to document categories. Since the weighting values are ranged between 0 and 1, the subspaces discovered by these algorithms are of soft. With k-means type methods (L.Jing, M.K.Ng, J.Xu, and J.Z.Huang. 2005), (C.Domeniconi, D.Gunopulos, and S.Ma. 2006), (L.Jing, M.K.Ng, and J.Z.Huang. 2007), (J.Z.Huang, M.K.Ng, H.Rong, and Z.Li. 2005), the algorithms iteratively group the documents into hard partitions.

1.1 Literature Review

In many applications, a document may include multiple topics and thus may relate to multiple categories at the same time, resulting in the requirement of fuzzy document clustering. On the other hand, due to its effectiveness in discovering clusters with overlapping boundaries, fuzzy clustering algorithms are able to reveal more accurate cluster structures within the document collections (Q.Wang, Y.Ye, and J.Z.Huang. 2006). In (J.Li, X.Gao, and L.Jiao. 2005), a feature-weighting algorithm combined with the fuzzy k prototypes algorithm was presented. The steps of feature weighting and data partitioning are separated in this algorithm. Recently, an algorithm named fuzzy W-k-means (Q.Wang, Y.Ye, and J.Z.Huang. 2006) was proposed. In this algorithm however, the dimensions are assigned with a uniform value for different clusters. Additionally, the fuzzy W-k-means (Q.Wang, Y.Ye, and J.Z.Huang. 2006) introduces two user defined parameters α and β, which are difficult to estimate in practice. In order to perform fuzzy clustering on high dimensional data, a new algorithm named FPC (Fuzzy Projected Clustering) was studied in our previous work (L.Chen, Q.Jiang, and S.Wang. 2008). This parameter free algorithm can generate "soft" partitions of the high dimensional data. In this paper, we will present some theoretical analysis of the FPC algorithm for document clustering. Furthermore, we will deal with a common problem of the existing methods, i.e., the robustness of the algorithms. It is widely known that the performances of the k-means type algorithms are highly dependent on their initial states. Unfortunately, most of the above methods make little of these in the context of high dimensional clustering that lead to unstable clustering results. We will improve the FPC to a new version R-FPC (Robust FPC), by introducing a new technique called R-Greedy to build a robust initial condition for the algorithm. The remainder of this paper is organized as follows. In Section 2, we describe the R-FPC algorithm. The R-Greedy technique is presented in Section 3. Section 4 presents the experiments and the performance results. The conclusions are given in Section 5.

2. The R-FPC Algorithm

Given a vector space model, the documents vectors may be presented by $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$, where $\mathbf{x}_i=(x_{i1}, x_{i2}, \ldots, x_{id})$ and d stands for the number of unique words in the model, n denotes the total number of documents, x_{ij} is the normalized word frequency of the j^{th} term in the document. We also call $\mathbf{x}_i$ a data point in the d-dimensional space. Let $\{C_1, C2\ldots. CK\}$ be the K document clusters, where C_k denotes a partition of document collections. The membership of $\mathbf{x}_i$ to C_k is denoted as u_{ki}.

In text subspace clustering, each category of documents is characterized by a subset of terms in the vocabulary that corresponds to a subset of dimensions in the data space. In this sense, we say that a cluster of documents is situated in a subspace of the original space. It is clear that a term may play unequally important roles to all the clusters. To measure such special correlations, an individual weighting value w_{kj} that ranges in [0,1] is assigned to $j^{th}(j=1,2,\ldots,d)$ term of cluster $C_k(k=1,2,\ldots,K)$, indicating how much the term is relevant to the cluster, with of more relevance, and larger weight. FPC finds the weight for each term from each cluster by minimizing the following objective function in the clustering process (L.Chen, Q.Jiang, and S.Wang. 2008):

$$J(C, V,W) = \Sigma^{K}_{k=1} \Sigma^{d}_{j=1}\Sigma^{n}_{i=1} u_{ki}w_{kj} / 2\sigma^{2}_{k} \ (x_{ij} - v_{kj})^2$$
$$- \Sigma^{K}_{k=1} d \ln \alpha_k / \sqrt{2\pi\sigma k} \Sigma^{n}i=1 \ u_{ki} + d \Sigma^{K}_{k=1} \Sigma^{n}_{i=1} u_{ki} \ln u_{ki} \quad (1)$$

subject to

$$\begin{cases} \Sigma^{d}_{j=1} \sqrt{w_{kj}} = 1, \ 0 \le w_{kj} \le 1, \ 1 \le k \le K \\ \Sigma^{K}_{k=1} u_{ki} = 1, \ 1 \le i \le n \\ \Sigma^{K}_{k=1} \alpha^{d}_{k} = 1 \\ \alpha_k \ge 0, \ u_{ki} \ge 0, \ \sigma_k > 0, \ 1 \le k \le K, \ 1 \le i \le n \end{cases}$$

We have built an extended Gaussian model for subspace clustering and derived the above objective function(referred to (L.Chen, Q.Jiang, and S.Wang. 2008) for more details). Here αk is the mixture coefficient of k^{th} Gaussian component, $\mathbf{v}_k = (v_{k1}, v_{k2}, \ldots, v_{kd})$ and σ_k denote mean and covariance of the kth Gaussian, respectively. V and W denote the mean and weight matrix for all the K clusters, respectively.

Since all inputs $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ are available, the learning of all the parameters via minimizing Eqn.(1) can be implemented by the Expectation Maximization(EM) algorithm in a batch way. The algorithm R-FPC is summarized as follows, which is based on our previous algorithm FPC (L.Chen, Q.Jiang, and S.Wang. 2008) and introduces a new procedure called R-Greedy to build a robust initial condition for the algorithm. The R-Greedy

method will be described in next section.

Algorithm 1 R-FPC

Input: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$,, K and a termination criterion ε

Output: U = $\{u_{ki}|k=1,2,\ldots,K;i=1,2,\ldots,n\}$ and the associated weights matrix W.

begin

1 Initialization

1.1 Let p be the number of iteration, p = 0;

1.2 Call R-Greedy to initialize the $V^{(0)}$ and $W^{(0)}$;

1.3 Set $u_{ki} = 1/K$ for k=1,2,. . . ,K and i=1,2,. . . ,n;

Set α_k and σ_k to an constant.

2 Repeat

2.1 Set p=p+1

2.2 Use Eqn.(3) to calculate U(p)

2.3 Use Eqn.(2) to calculate W(p)

2.4 Update α_k(k=1,2,. . . ,K) using Eqn.(5)

2.5 Update σ_k(k=1,2,. . . ,K) using Eqn.(6)

2.6 Use Eqn.(4) to calculate V (p)

until $| \| V(p) \| \infty - \| V(p-1) \| \infty | < \varepsilon$

and $| \| W^{(p)} \| \infty \| W^{(p-1)} \| \infty | < \varepsilon$

3 Output U(p) and W(p)

end

Starting from the initial conditions, the algorithm updates U, W, V, α_k and σ_k (k=1,2,. . . ,K) iteratively. The process is repeated until no significant changes can be made for the partitions. In each step, the optimal values of parameter are computed to minimize J on the assumption that other parameters are fixed. Using Lagrange multiplier technique, we can solve the feature weight w_{kj} by

$$w_{kj} = (1/X_{kj} + \delta)^2 / (\Sigma^d_{j=1} 1/X_{kj} + \delta)^2 \quad (2)$$

with

$$X_{kj} = \Sigma^n_{i=1} u_{ki}(x_{ij} - v_{kj})^2.$$

Following FWKM (L.Jing, M.K.Ng, J.Xu, and J.Z.Huang. 2005), (J.Z.Huang, M.K.Ng, H.Rong, and Z.Li. 2005), to ensure that the denominator of Eqn.(2) is always larger than 0, we adjust the denominator by adding an additional factor $\delta = 1 / nd \, \Sigma^n_{i=1} \Sigma^d_{j=1} (x_{ij} - o_j)^2$ (L.Jing, M.K.Ng, J.Xu, and J.Z.Huang. 2005), where o_j is the mean feature value of the entire data set. The proof of Eqn.(2) can be found in (L.Chen, Q.Jiang, and S.Wang. 2008). Similarly, the membership matrix U in each iteration is updated by

$$u_{ki} = (\Sigma^K_{l=1} \alpha l/\sigma l \exp (-1/2d\sigma^2 l \, \Sigma^d_{j=1} w_{lj} (x_{ij} - v_{lj})^2))^{-1}$$
$$\times \alpha_k/\sigma k \exp (-1/2d\sigma^2_k \ \Sigma^d_{j=1} w_{kj} (x_{ij} - v_{kj})^2) \quad (3)$$

The above measurement of fuzzy membership degree is in an exponential type and is based on the weighted Euclidean distance, which is quite different from the one used in the FCM-based algorithm, such as the classical FCM (H.Sun, S.Wang, and Q.Jiang. 2004) and the newly designed algorithm fuzzy W-k-means(Q.Wang, Y.Ye, and J.Z.Huang. 2008). The means of Gaussian, i.e., the cluster center V, can be calculated by

$$v_{kj} = \Sigma^n_{i=1} u_{ki} x_{ij} \ / \Sigma^n_{i=1} u_{ki} \quad (4)$$

The Eqn.(4) is the same as the one defined in FCM (H.Sun, S.Wang, and Q.Jiang. 2004) (in case of the fuzzifer equals to 1). Fix U, V and W, we can derive the following equations to update α_k and σ_k:

$$\alpha_k = (u_{k+}/n)1/d \quad (5)$$

$$\sigma^2_k = 1 / d\Sigma^n_{i=1} u_{ki} \ \ \Sigma^d_{j=1} w_{kj} \ \ \Sigma^n_{i=1} u_{ki}(x_{ij} - v_{kj})^2 \quad (6)$$

It can be seen that the R-FPC is an extension to the FCM algorithm by adding multiple steps to estimate the parameters of the clustering model. Therefore the algorithm is able to converge within a finite number of iterations. The time complexity is O(hndK), where h is the total number of iterations.

2.2 The R-Greedy Method

The R-Greedy aims to provide a method for choosing the stable K cluster centers and their initial subspaces for RFPC. Most existing algorithms only consider the selection of initial K cluster centers by random selection (L.Jing, M.K.Ng, J.Xu, and J.Z.Huang. 2005)(L.Jing, M.K.Ng, and J.Z.Huang. 2007)(J.Z.Huang, M.K.Ng, H.Rong, and Z.Li. 2005)(Q.Wang, Y.Ye, and J.Z.Huang. 2008) or the Greedy technique (C.Domeniconi, D.Gunopulos, and S.Ma. 2006). For example, the Greedy technique choose the first center random, and selects the others such that they are far from one another, and from the first chosen center. It is important to remark that such technique measures the distance between data points by considering all features of the space. In high dimensional spaces, the data are inherently sparse; the distance between every pair of points is almost the same for a wide variety of distance functions. As we can see from the experimental results (presented in next section), using these traditional techniques for document clustering would lead to instability and poor accuracy of the clustering results.

We also argue that the subspaces where the initial clusters are situated should be taken into account at the initialization stage. It is because the true distances between data points will be distorted by noisy attributes in the high dimensional data space. Virtually all-existing soft subspace clustering algorithms evenly set the feature weights with all entries equal to a constant. The R-Greedy is an extension of the Greedy technique by considering these special characteristics of high dimensional data clustering, as follows.

Algorithm 2 R-Greedy

Input: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$, and K

Output: $V^{(0)}, W^{(0)}$.

begin

1 Initialization

1.1 Use Eqn.(8) to choose the first cluster center $\mathbf{v}_1$;

1.2 Use Eqn.(7) to calculate w_{1j} for j=1,2,. . . ,d;

2 For k=**2 to** K **do**

2.1 For each point $\mathbf{x}_i \notin \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{k-1}\}$, calculate $dist(\mathbf{x}_i) = min_{l=1,2,\ldots,k-1} \Sigma^d_{j=1} w_{lj}(x_{ij} - v_{lj})^2$

2.2 Choose the point $\mathbf{x}_l$ as the k^{th} cluster center using the following rule:

$l = argmax_{i=1,2,\ldots,n} dist(\mathbf{x}_i)$

2.3 Use Eqn.(7) to calculate w_{kj} for j=1,2,. . . ,d;

3 Output V as V (0) and W as W(0)

end

There are two major extensions in R-Greedy comparing with the traditional Greedy technique. Since the cluster centers with random selection may result in unstable clustering results, especially on the high dimensional data, we choose a determinable point as the first center at first. Secondly, R-Greedy searches other well-scattered centers using the weighted Euclidean distance function, which calculates the distances between data points in individual subspaces. The initial subspaces for all chosen cluster centers are computed based on Eqn.(2). In particular,

$$w_{kj}^{(0)} = (1/X_{kj}^{(0)} + \delta)^2 / (\Sigma^d_{j=1} 1/X_{kj}(0) + \delta)^2 \quad (7)$$

with

$$X_{kj}^{(0)} = \Sigma^n_{i=1} (x_{ij} - v_{kj})^2$$

The task of step 1.1 is to choose a point from the dataset that candidates to be one of the centers of the underlying document clusters. Since we use tf representation for documents, each entry of the data is proportional to the term frequency and has been normalized in range of [0,1]. Therefore, the length of a vector in such representation is able to measure the relevance degree of the corresponding document with its topic, to some extent. Based on this observation, we can choose the first cluster center according to the following rule:

$$\mathbf{v}_1 = \text{argmax}_{\mathbf{x}1,\mathbf{x}2,\ldots,\mathbf{x}n} \Sigma^{d}_{j=1} x^2_{ij} \quad (8)$$

The time complexity of R-Greedy is O(ndK). More importantly, the R-Greedy can always generate determinable initial conditions for the clustering algorithm. Since fuzzy clustering is generally better than hard clustering at avoiding local minima, using R-Greedy the R-FPC can achieve the robust clustering results with better performance than existing text subspace clustering algorithms..

3. Proposed System

The proposed system is designed to perform the document clustering using the semantic analysis mechanism. The ontology is used for semantic analysis. The fuzzy logic technique is used for the clustering process. The fitness analysis is performed to verify cluster accuracy. The sub space clustering scheme is used in the system. The document attributes are collected and grouped with relevancy. The similarity measurement is estimated on the sub space model. The sub space similarity model reduces the computation complexity and increases the accuracy. The sub space model also reduces the process time.

The clustering system is developed as a stand alone tool. The document preprocessing and clustering operations are handled by the system. The system uses the text documents for the clustering process. The text documents are collected from the benchmark datasets provided in UCI machine learning repository. The system is divided into four major modules. They are Document preprocessing, Term cluster, Semantic Cluster and Performance analysis.

The document-preprocessing module is designed to convert the documents into structured data sets format. The term cluster module is used to perform the document clustering using the term weights. The semantic clustering module is designed to cluster the documents using semantic weights. The performance analysis module is designed to analyze the cluster accuracy and process time. The system uses the Oracle relational database system as back end.

3.1 Document Preprocess

The documents are maintained in text file format. The contents of the documents are parsed and converted into the vector space model. The stop word elimination and stemming process are used to reduce the vector size. The system maintains a stop word repository. The stop words in the documents are removed using the repository. The stemming process analyzes the suffix value for the terms. The base term is extracted using the stemming process. The porter-stemming algorithm is used in the system. The document details are updated into the database. The system also updates the term list into the database.

3.2 Term Cluster

The system performs two types of clustering operations. They are term clustering and the semantic clustering. The term clustering task is performed using the term weights. The term frequency is estimated and updated into the database. The term frequency and inverse document frequency are calculated for each term. The term weights are used for the similarity measurement process. The fuzzy clustering scheme is applied on the sub space of the term collection. The term weights are used for the comparison process. The term cluster requires high vector size for the clustering process.

3.3 Semantic Cluster

The semantic clustering is performed with the term relationship based comparison. The term cluster does not consider the term relationship. The semantic cluster uses the term relationship for the clustering process. The ontology is used to maintain the relationship for the term collection in a domain. The terms are maintained with synonym, meronym and hypernym relationships. The terms are analyzed with the ontology collections. The term category is used for the weight estimation process. The semantic weight is estimated for each concept. The clustering process uses the semantic weights.

3.4 Performance Analysis

The performance analysis module is designed to analyze the performance of the term clustering and semantic clustering techniques (Figure1). The memory, process time and accuracy metrics are used for the performance analysis. The memory requirement for each clustering is analyzed. The accuracy is estimated using the fitness function.

4. Experiments and Performance Results

The text documents are denoted as unstructured databases. It is very complex to group the text documents. The document clustering requires a preprocessing task to convert the unstructured data values into a structure one. The documents are large dimensional data elements. The dimension is reduced using the stop word elimination and stemming process. The ontology fuzzy document is the process of extracting the frequent and popular contents of the text document collection. The document grouping tasks require the content relationship factors. The semantic analysis is the technique that uses the term and its relationship with a collection of terms. The relationships are represented as synonym, meronym and hypernym. The system is implemented to perform fuzzy text document grouping with the support of semantic analysis. Table1 shows the analysis of term cube versus semantic cube. The benchmark document collection is selected as the testing environment for the system.

The system is tested with benchmark document collection from 20 newsgroup dataset. Initially the documents are updated to the database with preprocessed information. The stopword elimination and stemming operations are performed in the preprocessor. All the document analysis operations are carried out on the database information. The porter-stemming algorithm is used in the system. The system is implemented to perform fuzzy text document grouping with the support of semantic analysis. Table1 shows the Memory Usage Analysis-3 Clusters K-means Vs Fuzzy, Table2 shows the process Time analysis-3 Clusters K-means Vs Fuzzy and Table3 shows the Fitness Point Analysis-3 Clusters K-means Vs Fuzzy.

5. Conclusion

Text clustering is about discovering novel, interesting and useful patterns from textual data. In this paper we have discussed how to introduce the method of building ontologies into unsupervised text learning in order to consider the text semantics in the preview of linguistics. The fuzzy document clustering uses the sub space-clustering model. The relevant attributes are used for the comparison process. The semantic analysis is used to reduce the vector size. The relevancy is also improved by the semantic analysis. The system can be enhanced with multi domain ontology to analyze documents with any domain. This also applied to distribute clustering on web document and in XML document. In Future work will consider the fuzzy clustering scheme under the direction of ontologies, after all, most of the documents simultaneously belong to more than one category. Furthermore, the method of calculating the term mutual information in this paper can be used to create the ontology in different field.

References

C.Domeniconi, D.Gunopulos, and S.Ma. (2006). “Locally adaptive metrics for clustering high dimensional data,” *Technical Report ISE-TR-06-04,* George Mason University, 2006.

H.Sun, S.Wang, and Q.Jiang. (2004). “Fcm-based model selection algorithms for determining the number of clusters,” *Pattern Recognition,* vol. 37(10), pp. 2027–2037, 2004.

J.Li, X.Gao, and L.Jiao. (2005). “A novel feature weighted fuzzy clustering algorithm,” *LNAI,* vol. 3641, pp. 412–420, 2005.

J.Z.Huang, M.K.Ng, H.Rong, and Z.Li. (2005). “Automated variable weighting in k-means type clustering,” *IEEE Transactions on Knowledge and Data Engineering,* vol. 27(5), pp. 657–668, 2005.

L.Chen, Q.Jiang, and S.Wang. (2008). “A probability model for projective clustering on high dimensional data,” *Proceeding of the IEEE ICDM,* pp. 755–760, 2008.

L.Chen, Y.Ye, and Q.Jiang. (2008). “A new centroid-based algorithm for text categorization,” *Proceeding of the AINAW,* pp. 1217– 1222, 2008.

L.Jing, M.K.Ng, and J.Z.Huang. (2007) “An entropy weighting kmeans algorithm for subspace clustering of high-dimensinoal sparese data,” *IEEE Transactions on Knowledge and Data Engineering,* vol. 19(8), pp. 1–16.

L.Jing, M.K.Ng, J.Xu, and J.Z.Huang. (2005). “On the performance of feature weighting k-means for text subspace clustering,” *Proceeding of the WAIM,* pp. 502–512, 2005.

Lifei Chen, Shengrui Wang and Qingshan Jiang. (2009). “A Robust Algorithm for Fuzzy Document Clustering”, 2009.

Q.Wang, Y.Ye, and J.Z.Huang. (2008). “Fuzzy k-means with variable weighting in high dimensional data analysis,” *Proceeding of the WAIM,* pp. 365–372, 2008.

S.B.Kotsiantis and P.E.Pintelas. (2004) “Recent advances in clustering: A brief survey,” *WSEAS Transactions on Information Science and Applications,* vol. 11(1), pp. 73–81.

Table 1. Process Time Analysis- 3 Clusters K-means Vs Fuzzy

S.No	Documents	K-means(msec)	Fuzzy (msec)
1	500	162	97
2	1000	336	196
3	1500	494	281
4	2000	658	372
5	2500	825	466

Description for the above table.

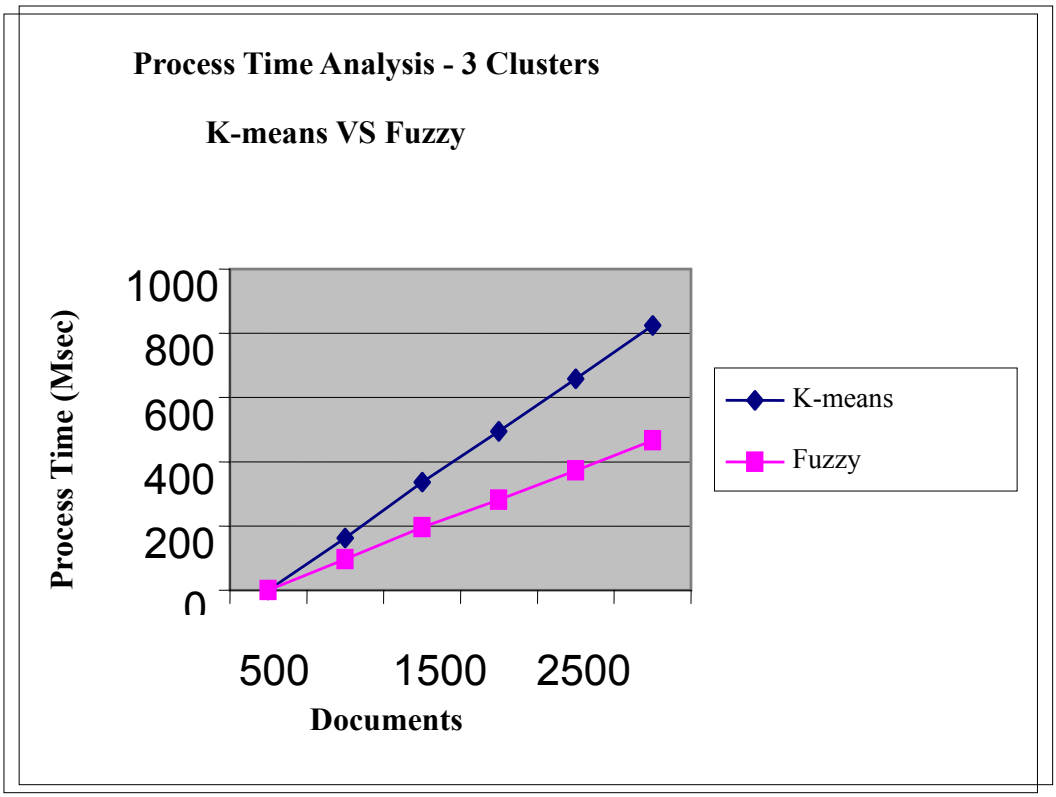


Table 2. Memory Usage Analysis – 3 Clusters K-means Vs Fuzzy

S.No	Documents	K-means (kb)	Fuzzy(kb)
1	500	9	12
2	1000	17	23
3	1500	26	34
4	2000	37	45
5	2500	45	54

Description for the above table.

ISSN 1913-1844 E-ISSN 1913-1852

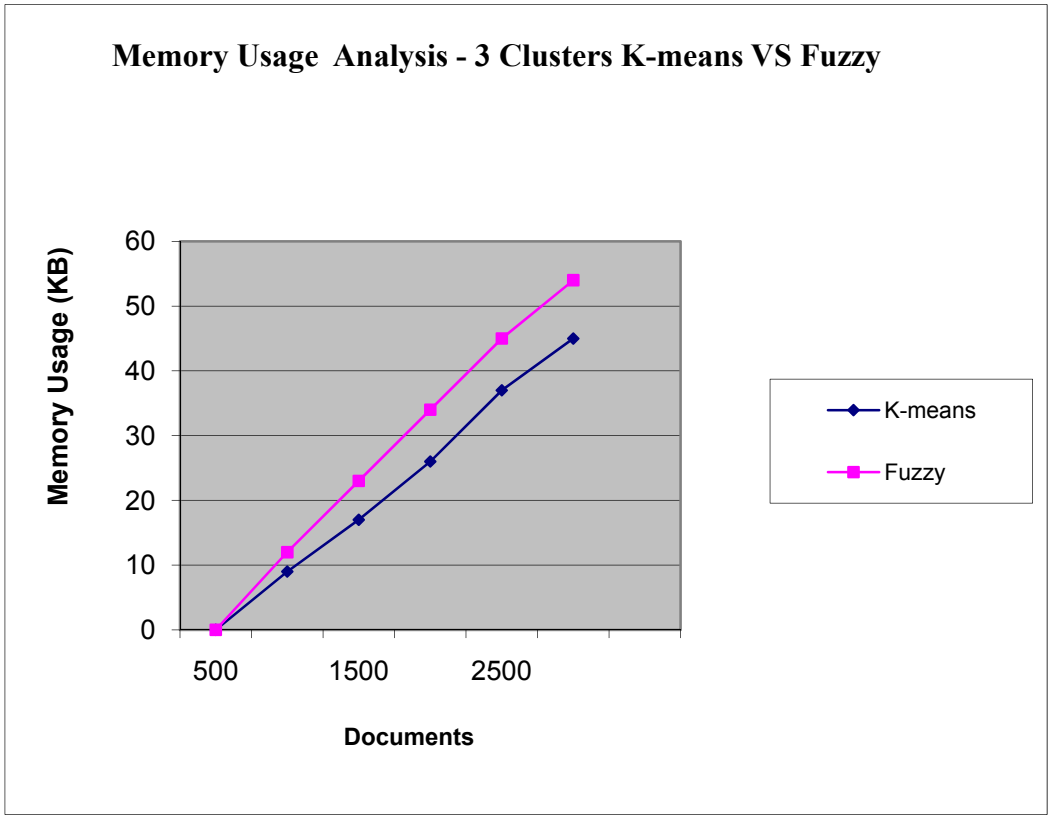
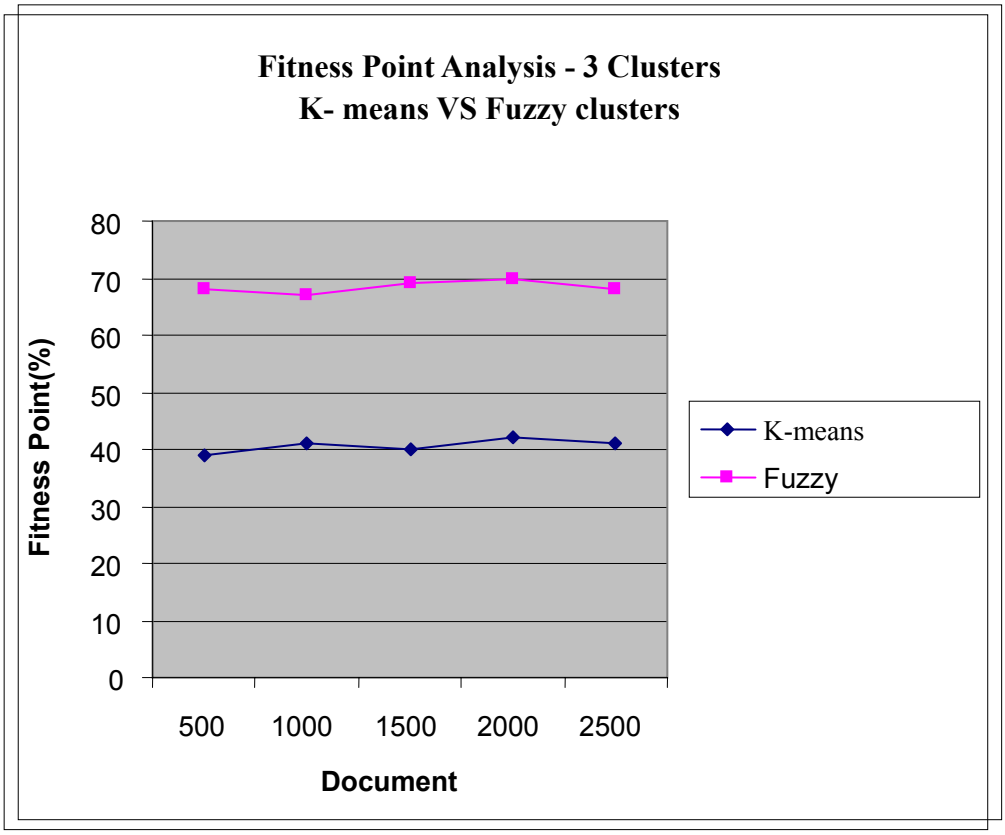


Table 3. Fitness Point Analysis-3 Clusters K-means Vs Fuzzy

S.No	Documents	K-means	Fuzzy
1	500	39	67
2	1000	41	68
3	1500	40	69
4	2000	42	70
5	2500	41	68

Description for the above table.



ISSN 1913-1844 E-ISSN 1913-1852