Overview of Clustering High Dimensionality Data using Hubness Algorithm

Nikita Dhamal
Department of Computer Science and Engineering
G H Raisoni Institute of Engineering
and technology for women
Nagpur, India
nikitadhamal@gmail.com

Antara Bhatttacharya
Department of Computer Science and Engineering
G H Raisoni Institute of Engineering
and technology for women
Nagpur, India
antarabhattacharya@raisoni.net

***Abstract*—-** High dimensionality data clustering can be seen in all fields these days and is becoming very tedious process. The important disadvantage of high dimensional data which we can give is that of curse of dimensionality . As the magnitude of datasets grows the data points become sparse and density of area becomes less making it difficult to cluster that data which further reduces the performance of traditional algorithms used for clustering . To rout these toils hubness based algorithms were introduced as a variation to the these algorithms which influences the distribution of the data points among the k-nearest neighbor. The hubness is an unguided method which finds out which points appear more frequently in the k-nearest neighbor than other points in the dataset. This paper discuss the ways of clustering algorithms using hubness phenomenon. One of the methods is based on condensed nearest neighbor which is performed iteratively on the order independent data. The next algorithm is hinged for fuzzy based approaches which performs better on uncertain data ie . partially exposed or incomplete data. The proposed algorithms are basically used for increasing the efficiency and increasing predicting accuracy of the system**.**

***Keywords*-** *clustering , high dimensional data , hubness , nearest neighbor*.

*****

I. INTRODUCTION

Clustering of data provides us with a way to group elements together such that elements of same group are of similar attributes or features .Based on the enactment of clusters the criteria for clustering changes. Clustering is often muddled with classification, but classification differs with clustering in a way that in clustering both classes and the objects included in clustering are already defined i.e. predefined. With the help of clustering techniques objects which are logically similar to each other are physically kept near to each other. According to [1] the various clustering algorithms are randomly sketched into 4 types namely: partitional algorithms, hierarchical algorithms, densitybased algorithms , and subspace algorithms. Out of these the subspace algorithms are basically used to cluster high dimensionality data. High dimensionality usually refers to a large number of attributes of the specified objects.When the dimensions of the data increases it leads to the curse of dimensionality which reduces the performance of the clustering algorithms. The curse of dimensionality refers to the problem of handling the data when the number of dimensions increases.

The concept of hubness is used to handle datasets containing high dimensional data points. Due to the increasing dimensionality of a data set sharing of the number of times a data point appears among the *k* nearest neighbors of other data points in the datasets becomes highly bevelled . As the dimensions of the data increases, the time needed for the execution increases and efficiency required goes on decreasing. The traditional machine learning algorithms and methods can be further modified to increase the accuracy and the efficiency of algorithms. The algorithms to be used are based on the k nearest neighbor technique of clustering. The shared neighbor algorithm will support to relate data of different clusters which in turn will provide better clustering. Density based approaches can also be added with the shared neighbor algorithm to provide more efficiency.

The fast condensed nearest neighbour algorithm is an advancement to condensed nearest neighbour algorithm which is used to reduce the dataset for classification based on some proptotypes. Here the learning speed of the algorithm is also considered which tells us that the algorithm should give good behaviour under all conditions. The Fuzzy Rough Nearest Neighbor algorithm is based on the classification that the the description of vector space is not proper. Due to this we get imperfect classification space resulting in the uncertainity of result. The FRNN algorithm can provide us with more accurate prediction of clustering result. The k-NN algoritms gives results which are satisfactory for high dimensional data. The applications of these clustering algorithms can be seen in various fields like text mining, text retrieval , classification image feature and many more.

Clustering algorithms	**Examples**
Partitional	k-means, k-medoids.
Heirarchical	Agglomerative,divisive.
Density based	DBScan.
Subspace	k-nn,Scaf

TABLE 1.TYPES OF CLUSTERING ALGORITHMS

This paper mainly concentrates on two clustering algorithms i.e. Fast condensed nearest neighbor and fuzzy rough nearest neighbor and the clusters formed due to the application of these algorithms.the main ain is to increase the performance of the algorithms when improper data is given.

The remainder of this paper is organized as follows: Section 2 surveys the different issues of various clustering algorithms related to the system. Section 3 presents a brief review of related work. Section 4 examines the proposed plan of the system

II. ISSUES ASSOCIATED WITH THE EXISTING SYSTEM.

The first issue is the Difficulty to implement for some of the data types. Take an example of colour representation or of geographical location .This is due to the fact that it relies on being able to get a quantitative result from comparing two items.

The second issue is that it can often get around by converting the data to a numerical form, for example converting colour to an RGB value, or geographical location to latitude and longitude.

Another disadvantage is that it becomes slow for large databases. This is mainly because each new entry is to be compared to all other entry. This can be sped up using data reduction.But this can further result in improper exposure of data.

We can investigate how well CNN manages to reduce the number of data points, in the presence of different quantities of random noise.To do this, we record what percentage of points were assigned each classification on each of three different data sets as we increased the number of random noise points.

III. RELATED WORK

Based on the above discussion current work can be divided into two categories: partial exposure of data and order independence of the data.

A. PARTIAL EXPOSURE OF THE DATA

The general idea behind clustering is to partition a given dataset into homogeneous subsets. One popular approach consists in finding a partition of the original space and assigning each data element to one of the clusters by means of a similarity function, which is often based on the Euclidean distance as a metric. Each cluster is then represented by **a** prototype, or cluster representative. The well-known fuzzy c-means algorithm is an example for such a clustering algorithm, m where in addition one allows each data element to belong to all clusters simultaneously, but to different degrees. In formal terms, assuming we have a data set

$$X = \{x_1, .., x_{|X|}\} \subset \mathbb{R}^n, n \in \mathbb{N}$$

the aim is to compute the prototypes P = ***{PI,. . . ,P$_{IPI}$)***
as a result of the following optimization problem:

$$J_m(X; U, P) = \sum_{j=1}^{|X|} \sum_{i=1}^{|P|} u_{i,j}^m d_{i,j}^2$$

B. ORDER INDEPENDENT CLUSTERING

Condensed nearest neighbor is the algorithm which is designed to reduce the data sets that are used for *k*-NN classification. It selects a set of prototypes *U* from the training data, such that 1NN with *U* can classify the examples almost as accurately as 1NN does with the whole data set.

The CNN rule is the original, and perhaps simplest, of many such methods, all of which attempt to extract a subset from an entire set of samples. The common idea of these algorithms is to execute a process iteratively to check the satisfaction of certain criteria for the current set of prototypes, and add or drop prototypes until a stop condition is met. The CNN algorithms focuses on the instance based learning. They have the advantage of extracting prototypes rapidly, since they adopt samples as prototypes and thereby avoid the rather costly computation of clustering.Given a training set *X*, CNN works iteratively:

1. Scan all elements of *X*, looking for an element *x* whose nearest prototype from *U* has a different label than *x*.
2. Remove *x* from *X* and add it to *U*
3. Repeat the scan until no more prototypes are added to *U*.

Use *U* instead of *X* for classification. The examples that are not prototypes are called "absorbed" points .

The condensed nearest neighbor rule retains the basic approach of the NN rule but uses only a subset of the training set of samples. This subset, when used as a stored reference set for the NN decision rule, correctly classifies all the samples belonging to the original training set. As the CNN method chooses samples randomly, internal rather than boundary samples are occasionally retained. The proposed scheme can yield significantly better accuracy than other instance-based data reduction methods. The concept of mutual nearest neighborhood is used to obtain a modified condensed training set. The algorithm, called FCNN rule, has some desirable properties. Indeed, it is order independent and has sub-quadratic worst case time complexity, while it requires few iterations to converge, and it is likely to select points very close to the decision boundary . Thus the Proposed FCNN algorithm tries to remove the drawbacks of CNN and improve efficiency of clustering.

IV. PROPOSED METHODLOGY

The basic purpose of the proposed system is to develop clustering algorithm which work on the order independent data as well as partially exposed data. As hubness defines better clustering on high dimensional data different algorithm based on the hubness concept have been implemented. These algorithms vary in the way by which they operate on various types of data and the output clusters we get when we use these algorithms.The following are the proposed plan of work:-

1) FCNN Module
2) FRNN Module.
3) APPLICATION OF ALGORITHM

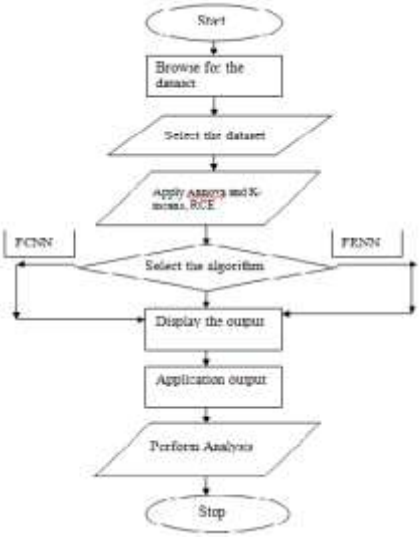

Fig1..Flowchart for above proposed scheme

Module 1 FCNN algorithm:-

In the first module before applying the FCNN algorithm we we apply the annova algorithm and k-means/RCE algorithm which will help to improve the clustering of data.

In the analysis of variance, it is assumed that different samples have equal variances, which is commonly called homogeneity of variance. The Levene test and Brown-Forsythe test can be used to verify the assumption. Suppose we have *k* samples of response data, where y_{ij}represents the value of ith observation (*i* = 1, 2, ...n_j) on the *j*th factor level (*j* = 1, 2, ..., *k*). The hypotheses of both Levene test and Brown-Forsythe test can be expressed as:

H_0: $\sigma_1^2 = \sigma_2^2 = \cdots = \sigma_k^2$

H_1: $\sigma_p^2 \neq \sigma_q^2$, for at least one pair (*p*, *q*), $1 \leq p, q \leq k$

Because 'time' is treated as a qualitative factor in the ANOVA decomposition preceding, a nonlinear multivariate time trajectory can be modeled.

Condensed nearest neighbor is a clustering algorithm designed to reduce the data set for *k*-NN classification space.It selects the set of prototypes *U* from the training data, such that 1NN with *U* can classify the examples almost as accurately as 1NN does with the whole data set. Given a training set *X*, CNN works iteratively:

I. Scan all elements of *X*, looking for an element *x* whose nearest prototype from *U* has a different label than *x*.
II. Remove *x* from *X* and add it to *U*
III. Repeat the scan until no more prototypes are added to *U*.

Use *U* instead of *X* for classification. The examples that are not prototypes are called "absorbed" points.

It is efficient to scan the training examples in order of decreasing border ratio. The border ratio of a training example *x* is defined as : $a(x) = ||x'-y|| / ||x-y||$

where ||*x*-*y*|| is the distance to the closest example *y* having a different color than *x*, and ||*x'*-*y*|| is the distance from *y*to its closest example *x'* with the same label as *x*.

The border ratio is in the interval [0,1] because ||*x'*-*y*|| never exceeds ||*x*-*y*||. This ordering gives preference to the borders of the classes for inclusion in the set of prototypes*U*. A point of a different label than *x* is called external to *x*. The calculation of the border ratio is illustrated by the figure on the right. The data points are labeled by colors: the initial point is *x* and its label is red. External points are blue and green. The closest to *x* external point is *y*. The closest to *y* red point is *x'*. The border ratio $a(x)=||x'-y||/||x-y||$ is the attribute of the initial point *x*.

Module 2 FRNN algorithm:-

Conventional fuzzy K-NN algorithm assigns an unlabelled pattern x to the class which appears the most among its k nearest labelled neighbors. The algorithm is described as follows. Conventional fuzzy NN algorithm:-

Part A: get the k nearest neighbors of the test pattern x
Let X={x1, x2,…, xn} be the set of already labelled data (training data), and C={c1,c2,…cc} is the result classification space. Let x be the unlabelled test data.
Input x; Set K, $1 \leq K \leq n$;
Set the iteration counter count=1;
For all xj ∈ X ($1 \leq j \leq n$) Do
Compute ‖x-xj‖
If ($i \leq K$)
include xj in the set of K-nearest neighbors and increase count by 1 else if (xj is closer to x than any previous nearest neighbor)
Begin Delete the farthest of the K-nearest neighbors
Include xj in the set of K-nearest neighbors .End
End For

Part B: approximate x by the k-nearest neighbors
For all cj ∈ C ($1 \leq i \leq c$) Do Compute ui(x)
End For

Part A is to choose some of the training data points that are similar to the test data point as its neighbors. Part B is to use the membership functions of the selected neighbors to compute the approximated membership of the test data point.

Module 3 Application of the algorithms in realtime:-

Sales forecast:

In the sales forecast the previous data of the company sales will be taken. Dividing the data into parts the clustering can be obtained. This will help to know at what time of the year the sales of the product were more as compared to other times.

This will help to predict the production quantity of goods the company produces thus increasing the overall profit ratio of the company as the sales will increase due to the prediction obtained based on the results of algorithm.

Weather forecast:

In the weather forecast the previous data of the weather conditions will be analysed. Through this analysis it can be predicted that at which time of the year the weather conditions will become worse so that proper action can be taken in advance to deal with it.

V. EXPERIMENTAL SETUP

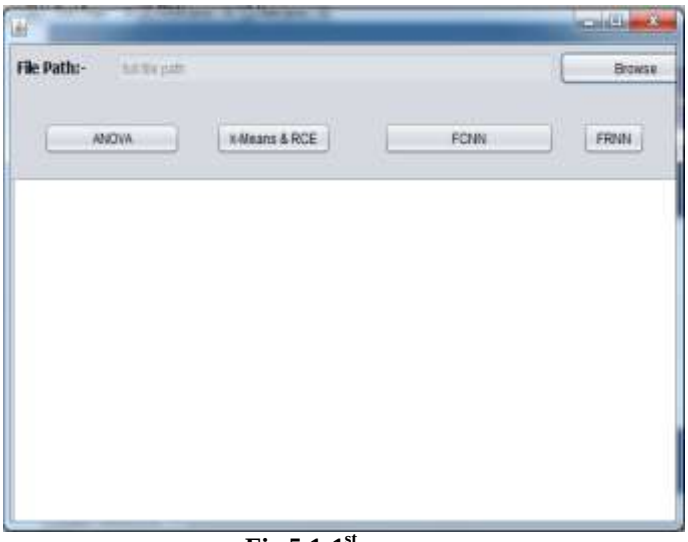

Fig 5.1:1st screen

 ISSN: 2321-8169
015 – 019

Fig 5.1 shows the first screen of the proposed system. From here we will browse for the file containing the data. In this the wine dataset from the UCI Machine Repository has been used. After that we will apply the Annova Algorithm to find out the variance in the datasets.

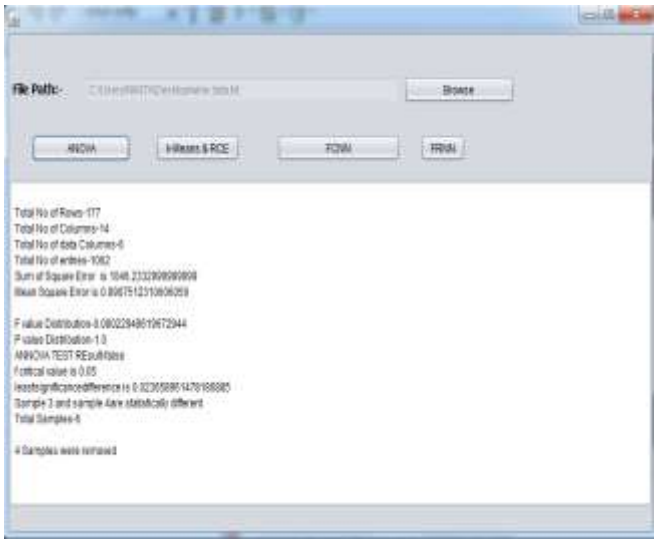

Fig 5.2 Application of Annova Algorithm

After the application of the annova algorithm we get the above output. Based on the varoius parameters specified we can caculate the actual variance in the dataset.Some of the datasets are removed either because the data cluster did not contain any items or it is stattisticaaly different.

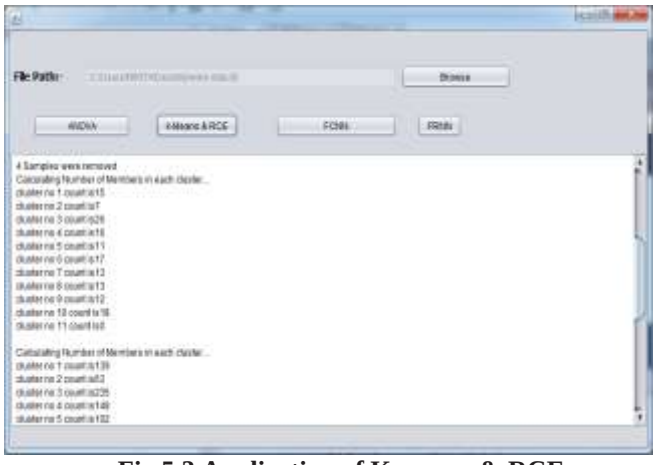

Fig 5.3 Application of K-means & RCE

The above fig.5.3 shows the output after applying the K-means & RCE algorithm. This will help to know how many rows and columns are there, total no. of data in the dataset, how much data a cluster contain individually and much more. i.e. full information about the data in the dataset is given.

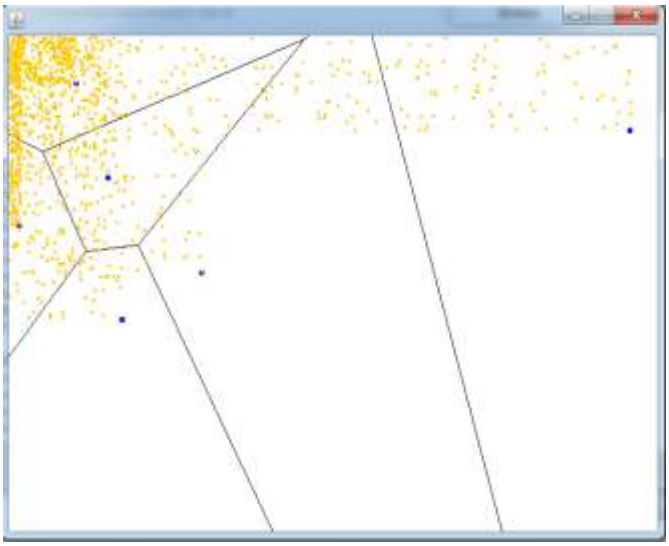

Fig 5.4: Application of FCNN Algorithm

After the FCNN algorithm is applied we get the following output. The FCNN algorithm is basically used to perform clustering on the order independent data. Better clustering performance can be seen due to this algorithm.

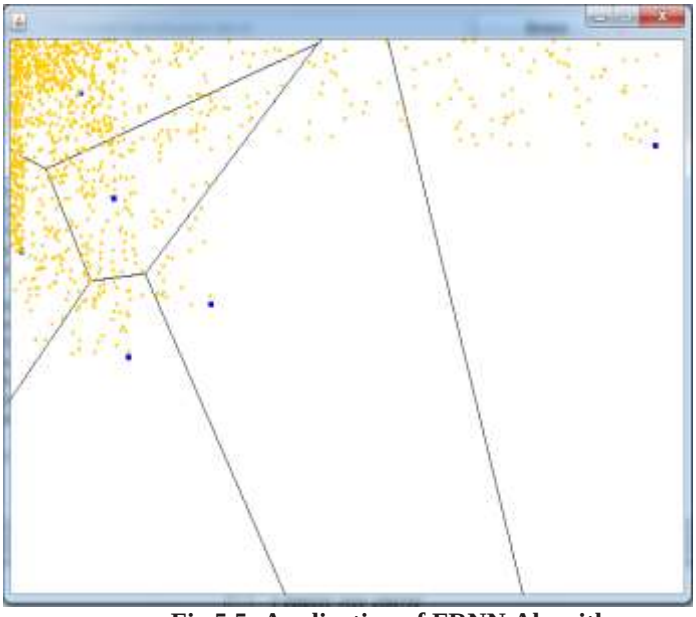

Fig 5.5: Application of FRNN Algorithm

After the FRNN algorithm is applied we get the following output. The FRNN algorithm is basically used to perform clustering on the partially exposed data. Better clustering efficieny can be seen due to this algorithm.

Finally after the application we can further give the analysis of the algorithm will be done based on the execution time needed by the algorithm, number of records used and finally the number of dimensions of the datasets. The number of dimensions increases the sparseness of the data which leads to the reduction in the efficiency of the algorithm. This analysis will help to know how the proposed system will be better than the existing system.

VI. CONCLUSION AND FUTURESCOPE

The concept of hubness can be used to cluster datasets containing high dimensional data. A lot of algorithms have been devised for clustering using the concept of hubness. The choice of the algorithm can be made depending upon the application for which we are going to use these algorithm. The fuzzy-based algorithms can work well when the data is either unbalanced or partially exposed as compared to other algorithms. The condensed nearest neighbour algorithm is best suitable for datasets which are order-independent. This is due to the fact that they make use of training datasets and approximate membership is found out which is done iteratively with each iteration providing better clustering. In this paper, it is shown that the different hubness algorithms can prove to be better than other types of clustering algorithms.

The major advantage of the said methods is their efficiency in clustering, when high dimensional data which is order independent and partially exposed. The FRNN shares the algorithmic simpleness with rest of the NN approaches we can see that FRNN method outperforms them by a comfortable margin, and that it is able to match with more methods including Support Vector Machines. Though the application shown cover a very small area but the many more areas like hospitals data and datawarehouses can also take the advantage of these algorithms

REFERENCES

[1] Nenad Tomasev, Milos Radovanovic, Dunja Mladenic, Mirjana Ivanovic."The Role of Hubness in Clustering High-Dimensional Data IEEE transactions on knowledge and data engineering, vol. 26, no. 3, march 2014.

[2] M. Radovanovi_c, A. Nanopoulos, and M. Ivanovi_c, "Hubs in Space: Popular Nearest Neighbors in High-Dimensional Data," J. Machine Learning Research, vol. 11, pp. 2487-2531, 2010.

[3] N. Toma_sev, M. Radovanovi_c, D. Mladeni_c, and M. Ivanovi_c," Hubness -Based Fuzzy Measures for High-Dimensional k-Nearest Neighbor Classification," Proc. Seventh Int'l Conf. Machine Learning and Data Mining (MLDM), pp. 16-30, 2011.

[4] N. Toma_sev, M. Radovanovi_c, D. Mladeni_c, and M. Ivanovi_c, "The Role of Hubness in Clustering High Dimensional Data," Proc. 15th Pacific-Asia Conf. Knowledge Discovery and Data Mining (PAKDD),Part I, pp. 183-195, 2011.

[5] N. Tomasev, M. Radovanovic, D. Mladenic, and M. Ivanovic, "A Probabilistic Approach to Nearest-Neighbor Classification: Naïve Hubness Bayesian kNN," Proc. 20th ACM Int'l Conf. Information and Knowledge Management (CIKM), pp. 2173-2176, 2011.

[6] D. Schnitzer, A. Flexer, M. Schedl, and G. Widmer, "Local and Global Scaling Reduce Hubs in Space," J. Machine Learning Research, vol. 13, pp. 2871-2902, 2012.

[7] N. Toma_sev and D. Mladeni_c, "Nearest Neighbor Voting in High Dimensional Data: Learning from Past Occurrences," Computer Science and Information Systems, vol. 9, no. 2, pp. 691-712, 2012.

[8] Sunita Jahirabadkar and Parag Kulkarni Clustering for High Dimensional Data: Density based Subspace Clustering Algorithms ,International Journal of Computer Applications (0975 – 8887) Volume 63– No.20, February 2013

[9] C. Ding and X. He, "K-Nearest-Neighbor Consistency in Data Clustering : Incorporating Local Information into Global Optimization ,"Proc. ACM Symp. Applied Computing (SAC), pp. 584-589, 2004.

[10] I.S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-Means: Spectral Clustering and Normalized Cuts," Proc. 10th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 551-556, 2004.

[11] H. Bian, L. Mazlack, Fuzzy-rough nearest neighbor classification approach, in: Proceedings of the 22th International Conference of the North American Fuzzy Information Processing Society (NAFIPS'03), Chicago, Illinois, USA, July 24–26, pp. 500–505.