*Research Article*

# Cluster Optimization for Boundary Points using Distributive Progressive Feature Selection Algorithm

**[1]Ch. Raja Ramesh, [2]K.V.S.N Rama Rao**
[1]Research Scholar, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, Guntur District, A.P., INDIA.chrajaramesh@gmail.com
[2]Professor,Department of CSE, Koneru Lakshmaiah Education Foundation,Aziz Nagar, Moinabad (M), Hyderabad,kvsnrr@klh.edu.in

**Article History**: Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

**Abstract:** A group of different data objects is classified as similar objects is known as clusters. It is the process of finding homogeneous data items like patterns, documents etc. and then group the homogenous data items togetherothers groupsmay have dissimilar data items. Most of the clustering methods are either crisp or fuzzy and moreover member allocation to the respective clusters is strictly based on similarity measures and membership functions.Both of the methods have limitations in terms of membership. One strictly decides a sample must belong to single cluster and other anyway fuzzy i.e probability. Finally, Quality and Purity like measure are applied to understand how well clusters are created. But there is a grey area in between i.e. *'Boundary Points'* and *'Moderately Far'* points from the cluster centre. We considered the cluster quality [18], processing time and relevant features identification as basis for our problem statement and implemented Zone based clustering by using map reducer concept. I have implemented the process to find far points from different clusters and generate a new cluster, repeat the above process until cluster quantity is stabilized. By using this processwe can improve the cluster quality and processing time also.

**Keywords**: Cluster, Boundary points, Feature, Cluster Quality

## 1. Introduction

Data Mining (DM) is an interdisciplinary skill used to read the unidentified facts from the old datasets[3]. DM procedures are very popular in different departments like civil engineering, mechanical engineering, electrical engineering and other branches of engineering in recent days because of different requirements for present situation. Some of them are: size of the data set increased and requires more memory because of technology advancements. Identifying hidden data with these datasets is conceivable with the DM Skills.

To implement machine learning statistics reduce dimensionality is the process to decrease the numberof unrelated and same type variables under consideration,[1] by getting a set of principal variables. It can be bifurcate into feature extraction and featureselection[19]in ML statistics. To reduce features, random variables cab be consider by getting principal variables.

CBCA algorithm is implemented for utilized the USPS hand written dataset[9] which is used to get the quality of result this data set is also high dimensional data set. Assessment is done between k-means and CBCA and improved the accuracy of the system. The deep learning algorithm took the vital role to improve the outcomes.[10]kNN algorithm also implemented for bags of words to find the particular word is there are not there.[12] Analysis of time complexity is revealed that FCM performs much faster than fuzzy method. Further, all internal processes and stability metric procedures of fuzzy clustering and all validity indexes of FCM are found to be within the limits

### 1.1 Feature Selection

Feature selection approach try to find a subset of the originalvariables (also called attributes or features). In this process three different strategies can be used one is filter for information gain, wrapper is used for accuracy and embedded is used to add or remove while constructing the model based on the predicted errors [11]. In some data analysis cases such as classification or regression can be done in the reduced space more exactly than the original data space. In measurements of Machine Learning, include see the different problems[16], in few cases, dataanalysis such as data regression or data classification can be done in the reduced space more accurately than in the original space.

In measurements of Machine learning quality choice is the way toward sleeting a subset of highlights (factors, indicators) for use in display development. Highlighted choice procedures are utilized for four reasons:

- Models can be improved to make simple translate by users/analysts.
- preparing in short time,

- Scourge of dimensionality can be avoided,
- Speculation can be improved by dimension over fitting.

### 1.2 Feature Extraction

Extraction of features is a process of dimentionality reduction by which group of non-processed data reduced to more convenient groups for processing. The main characteristics of these huge data sets are a large number of processed variables that requires a lot of calculating resources to process. The main aim of extraction feature is the name for methods that combine and /or select variables into features, efficiently reduce the amount of data that must be handled, while still truthfully and entirely describing the original data set.

Highlights of feature extraction is a universal term for policies for constructing blends of the issues to get around these issues while still showing the data with adequate correctness. All the outputs can be enhanced and utilizing established arrangements for secondary highlights normally operated by a specialist[2]. This type of process is called include building. [6]In some cases usage the dimentionality reduction methods also. Some of the dimentionality reduction techniques are

1. Independent Component analysis
2. Kernel PCA
3. Latent Symantec Analysis
4. Principal component analysis
5. Partial least squares
6. Nonlinear dimensionality reduction etc.

Convert the required data from huge dimensional space to a space of fewer dimentions. In principal component analysis (PCA) data conversion may be sequential, but many nonlinear dimentionality reduction methods also exist.[4][5] For multidimensional data, tensor representation can be used in dimentionality reduction through subspace learning multilinker.

### 1.3 Feature Support Count

Feature support Count allows you to observer the number of features in the map based on subtype's rand feature classes respectively. First final number is given for each feature class after that for each subtype. The grand total for all the features in the process (map) is exhibited at the bottommost of the window. The total feature support count provides a portrait of the features that are presently loaded in the map. Feature class's waysare listed in the final Feature Count Window matches the table of contents and each feature class can be extended or warped to view amounts for distinct subtypes.

### 1.4 Euclidean distance

To measure the distance between two points Euclidean distance metric is took the major role, at the same time easily measure the data by using ruler for two and three dimensional spaces also. Sometimes Euclidean will also be selected in clustering [11].

### 1.5 Strength and Limitation of Existing KNN

There is a theoretical guarantee that with a huge dataset and large values of k, you're going to get good results from nearest neighbor learning. Nearest neighborhood methods can be lousy when p (the number of variable) is large because of the curse of dimensionality. In high dimension, it's really difficult to stay local. The main limitation of

theKnn is to make each prediction scan the entire training data set is very slow. To avoid this program we are going to implement MapReduce method by using Knn relief.

## 2. Related Work

### 2.1 Theoretically Optimal Feature Selection

The "optimal feature selection" framework [7], initially, places a sound theoretical foundation for the selecting features are the main task. Based on the surviving data theory, this framework describes the optimality for set of

features within the sense that it retains the foremost quantity of data needed for modeling the dependence between the input variables (features) and output variable (label) within the reduced-dimensional space.

Let T(x) denote the illustration of x when the spatial property reduction outlined by T this framework needs that the posterior p(y|T(x)) be as shut as attainable to the first one p(y|x)

### 2.2 Feature Weighting Relief

The processingissue of combinational examine is often some extent to be improving by employing a feature weighting strategy [3]. By using these feature weights consider as real-valued numbers rather than binary ones enables the utilization of some well-established optimization techniques and, thus, it allows for implementation of efficient algorithmic. Among the usual feature weighting algorithms, the RELIEF algorithm[4] is taken into account one among the foremost successful ones thanks to its simplicity and effectiveness[8]. Algorithm pseudo code is presented on reference [4]. The key idea of RELIEF is to iteratively evaluationof feature weights consistent with their ability to distinguish between neighboring patterns. In each iteration, a pattern x is randomly selected then two nearest neighbors of x are found, one from an equivalent class.

### 2.3 Feature Relief Algorithm for Bio-informatics

Yijun Sun et al.[3] have applied feature relief algorithm in Bioinformatics domain in two stages. **First**, in algorithmic features, preliminary from a new clarification of RELIEF, we put forward a set of feature weighting algorithms. The efficiency of those procedures, in terms of solution quality and computational proficiency, is experimentally established on a wide variety of data sets. Considering the augmented demand for analyzing data with large feature dimensionality in some developing domains such as bioinformatics, we expect widespread usage of these algorithms in these applications.

**Second**, in theoretical aspects, that paper wasproviding a new direction of feature selection research in addition to providing some new algorithms. Feature selection plays a critical role in machine learning. Yet, as opposed to classifier design, it still to date lacks rigorous theoretical treatment. This is fundamentally due to the trouble in defining an objective function that can be simplyimproved by some well-established optimization techniques. It is principally true for wrapper methods that use a nonlinear classifier to evaluate the goodness of selected feature subsets. The crisp divider of a feature set and the nonlinearity of a classification function make the resulting objective function non convex and even non differentiable. For this reason, greatest feature selection algorithms trust on empirical search. The I-RELIEF algorithms has a clearly defined objective function and can be solved through numerical analysis instead of combinatorial search and, thus, presents a promising direction for a more rigorous treatment of feature selection problems.

Sai Prasad et al. implementedCurse of dimentionality is the most serious downside of data in microarray as it has more number of attributes (features)[13]. This leads to disheartened computational stability. In microarray data analysis, identifying more relevant features required full attention. Most of the researchers applied two stage strategies for gene expression data analytics. In first stage, feature selection or feature extraction is employed as a preprocessing step to pinpoint more prominent features[17]. In second stage, classification is applied using selected subset of features. Based on this I have I applied clustering.

Manikandanet al. proposed [14] new type of clustering technique is KF represents combination of K-means and Fuzzy C-means algorithm. Here they are calculated the quality in terms of purity,entropy,recall and precision metrics [15].

### 2.4. kNN Relief Algorithm Implementation Using Map Reducers

There is No single method gives accurate results or avoid the practice, depend upon a single method of result. Because of this might not fit all sorts of data. Computing and space complexity also are available account when affect large data sets and data streams. Thus in any aspect selection of quite one method and aggregate the results or use the bulk voting of these methods.

This existing system uses ensemble approach andalso having some more capabilities to handle with large and really high dimensional data sets. Those are, make the algorithm as parallel, distributed and evolutionary. Parallelism is achieved through concurrent programming to completely utilize the CPU with the support of core processors. Distributed nature is achieved through MapReduce based implementation and eventually genetic algorithm is employed as evolutionary computing method to

automate the choice process without manual intervention in parameter tuning and cluster analysis as illustrated in fig. 1.

More over consequences are aggregated from all of the methods by selecting the coming together of features generated from various methods. These features successively applied to fuzzy clustering algorithm and evaluate the cluster quality. This procedure is recurrent till final set of related features are selected. This is often a onetime process. Once final set of features are selected and every one the opposite features are eliminated computation, reprocessing and space complexity are going to be reduced and also any clustering algorithm not only fuzzy clustering gives good results [2].

We use two sorts of dimensionality reduction techniques. One is non-linear based kernel functions and other is only statistical approach. Technically these two techniques are fully diversified methods. Thus more relevant features which are slot in all aspects are only selected with this approach. This approach is represented through the following model.

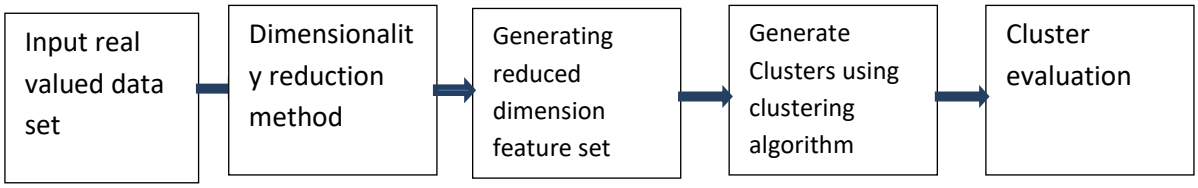


Fig:1 Existing Model for feature selection using map reduce approach

Based on the existing dimensionality reduction methods map reducer method is also one option to implement and get the better results. Everyone knows Knn is one of the best algorithms to identify nearest neighbors for normal data sets. If we implement along with map reducers it can use it for any type of data sets. In this paper[1] we implement Knn feature selection algorithm to get the better results for high dimensional data it is very simple by using existing java programming language with RMI. If the same apply for very high dimensional data and big data it may not be support but if we increase the number mappers in program it will work for very high dimensional data. It is very simple and useful to implement dimensionality reduction with efficient process.

## 3. Proposed System

In general, most of the clustering methods are either crisp or fuzzy and moreover member allocation to the respective clusters is strictly based on similarity measures and membership functions. Both of the methods have limitations in terms of membership. One strictly decides a sample must belong to single cluster and other anyway fuzzy i.e probability. Finally, Quality and Purity like measure are applied to understand how well clusters are created. But there is a grey area in between i.e. *'Boundary Points'* and *'Moderately Far'* points from the cluster centre. Boundary points are placed in between 2 cluster boundaries and moderately far points are having decent distance from cluster centre, means technically they are not tightly coupled with the cluster. To handle these kinds of scenarios this paper introduced a novel approach by incorporating 'Zone based approach' to further fine tune the clustering accuracy by handling boundary and moderately far points.

Following two diagrams fig.2 is exiting system with clusters along with their data points, in this red points are represented cluster centers black points are general points which is nearby center and yellow points are represented far from cluster center. Sometimes these points may have some difference with near points. To avoid this ambiguity we are proposed the new method to collect all boundary points from current cluster and nearby clusters. Fix the cluster center among these points then generate the new cluster. This process will continue until some stabilized clusters are generated,which is shown in fig. 3.

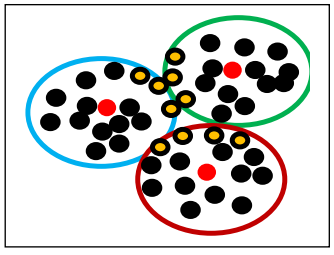

Fig. 2 Existing Method

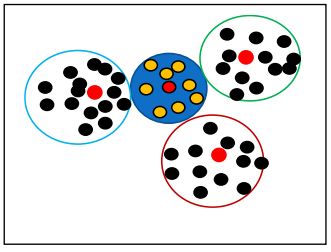

Fig. 3 Proposed Method

### 3.1 Model Building

Boundary points are placed in between the cluster boundaries and moderately far points are having decent distance from cluster centre, means technically they are not tightly coupled with the cluster. To handle these kinds of scenarios this paper introduced a novel approach by incorporating 'Zone based approach' to further fine tune the clustering accuracy by handling boundary and moderately far points as shown in fig. 4 To implement this process we have chosen the RMI environment from java programming. By using this environment implement the map-reducers for parallel processing to reduce the dimentions and finding feature support count.

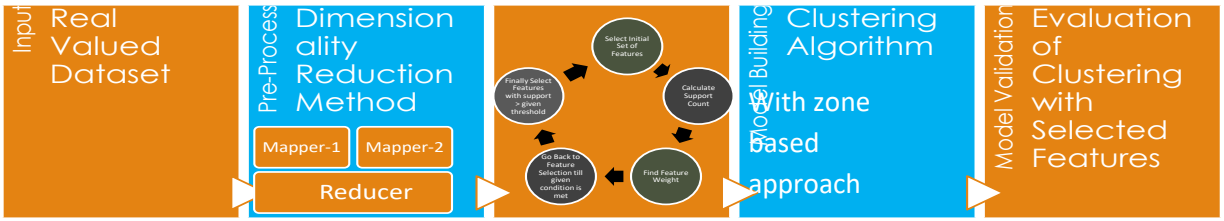


**Fig.4 Proposed System Model**

Following algorithm 1steps represents the proposed system.

0.Partition high dimensional input file into multiple files using vertical partition and place them in input folder.

1.Setup project properties such as thresholds, input, output folders

2.Build Map-Reduce Environment using RMI

2.1 Build One Mapper for each input file

2.1.1 Build Dataset at each mapper from respective mappers input file

2.1.2 Find Min and Max of each feature

2.1.3 Find normalized dataset

2.1.4 Execute Knn+Relieff Algorithm for Feature Selection

2.1.5 Build Dataset with selected features

2.2 Build Reducer by reading all the datasets generated from all mappers

2.2.1 Build the dataset at reducer with union of features and data instances extracted from above reduced data sets from mappers

2.2.2 Again apply kNN+ReliefF Algorithm on this dataset

2.2.3 Build Reduced Dataset with selected features

3 Save the selected features in a list

4. Repeat Steps 2.1,2.2 and 3 for given number of times (from properties file) (Evolutionary computing step)

5. Find the frequency of every feature after all repetitions

6. Find the support of each feature or dimension.

7. Mark the features with support > given threshold.

8. Build the dataset from finally marked features.

9. Apply Clustering on dataset with reduced features.

9.1. Find the boundary points from neighboring clusters.

9.2. Increase the cluster count.

9.3. Find the new cluster centers then repeat step 9 until get stabilized clusters.

10. Find the quality of clusters.

11. Build the dataset with all the features/dimensions.

12. Apply Clustering on whole dataset.

13. Find the cluster quality.

14. Compare the quality of these two methods.

15. *Find the time complexity for both distributed map-reduce frame work for comparison and Boundary clustering approach*

Breast cancer (BC) is one of the most common cancers among women worldwide, representing the majority of new cancer cases and cancer-related deaths according to global statistics, making it a significant public health problem in today's society. We have taken breast Cancer data set to generate the clusters. This data set contains nearly 1000 instances and 56 attributes. Table 1 shown two instances of the data set.

**Table1. Sample Data set**

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 575.7295 | 574.1896 | 581.24 | 573.4299 | 572.8094 | 570.3017 | 572.3299 | 575.8862 | 574.86 | 566.8223 | 212.5535 | 50.02681 | 36.79449 |
| 576.5099 | 574.4829 | 582.12 | 574.2 | 573.2495 | 571.0099 | 572.7373 | 576.0775 | 575.3855 | 567.0693 | 215.2113 | 50.02668 | 36.86074 |

## 6. Result Analysis

In this model we have tested with four algorithms represented as method1 to method 4. Method 1 which mentioned in the followingtables. Method 1 represents clustering without dimensionality reduction technique and without refined clustering (not consider the boundary points), Method 2 represents clustering with dimensionality reduction technique and without refined clustering,Method 3 represents clustering without dimensionality reduction technique and with refined clustering (not consider the boundary points), Method 4 represents clustering with dimensionality reduction technique and with refined clustering. These four methods are tested with different threshold values measure and tabulated in the tables 2 to table 10.

**Table 2. Results with threshold value 0.45**

| | Method Description | Refined Clustering | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0.45 | 56 | 3 | 1.75147044 | 0.016 |
| Method-2 | Clusters With Feature Selection | FALSE | 0.45 | 20 | 4 | 1.886187771 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0.45 | 56 | 6 | 1.343781746 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0.45 | 20 | 9 | 1.493756614 | 0 |

**Table3. Results with threshold value 0. 5**

| | Method Description | Refined Clustering? | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0.5 | 56 | 4 | 1.94960699 | 0 |
| Method-2 | Clusters With Feature Selection | FALSE | 0.5 | 20 | 5 | 1.903667629 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0.5 | 56 | 6 | 1.333314815 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0.5 | 20 | 9 | 1.441305115 | 0 |

**Table 4. Results with threshold value 0. 525**

| | Method Description | Refined Clustering? | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0.525 | 56 | 6 | 2.094532266 | 0 |
| Method-2 | Clusters With Feature Selection | FALSE | 0.525 | 20 | 5 | 1.881757576 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0.525 | 56 | 9 | 1.638883079 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0.525 | 20 | 11 | 1.559173554 | 0 |

**Table 5. Results with threshold value 0. 55**

| | Method Description | Refined Clustering? | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0. 55 | 56 | 7 | 2.295976988 | 0 |
| Method-2 | Clusters With Feature Selection | FALSE | 0. 55 | 20 | 5 | 1.881757576 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0. 55 | 56 | 10 | 1.941727665 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0. 55 | 20 | 11 | 1.559173554 | 0 |

**Table 6. Results with threshold value 0.6**

| | Method Description | Refined Clustering? | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0.6 | 56 | 11 | 2.563891423 | 0.016 |
| Method-2 | Clusters With Feature Selection | FALSE | 0.6 | 20 | 10 | 2.274003219 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0.6 | 56 | 11 | 2.563891423 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0.6 | 20 | 15 | 2.021754669 | 0 |

**Table 7. Results with threshold value 0.625**

| | Method Description | Refined Clustering? | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0.625 | 56 | 16 | 2.669878027 | 0 |
| Method-2 | Clusters With Feature Selection | FALSE | 0.625 | 20 | 14 | 2.460611239 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0.625 | 56 | 16 | 2.669878027 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0.625 | 35 | 20 | 2.460611239 | 0 |

**Table 8. Results with threshold value 0. 65**

| | Method Description | Refined Clustering? | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0.65 | 56 | 15 | 2.617670138 | 0 |
| Method-2 | Clusters With Feature Selection | FALSE | 0.65 | 20 | 14 | 2.460611239 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0.65 | 56 | 15 | 2.617670138 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0.65 | 20 | 14 | 2.460611239 | 0 |

**Table 9. Results with threshold value 0. 725**

| | Method Description | Refined Clustering? | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0.725 | 56 | 28 | 2.810338389 | 0 |
| Method-2 | Clusters With Feature Selection | FALSE | 0.725 | 20 | 24 | 2.759798535 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0.725 | 56 | 28 | 2.810338389 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0.725 | 20 | 24 | 2.759798535 | 0 |

**Table10 Results with threshold value 0. 75**

| | Method Description | Refined Clustering? | Threshold | NoD | No of Clusters | Quality | Time (in sec) |
|---|---|---|---|---|---|---|---|
| Method-1 | Cluster Without Feature Selection | FALSE | 0.75 | 56 | 28 | 2.810338389 | 0 |
| Method-2 | Clusters With Feature Selection | FALSE | 0.75 | 20 | 24 | 2.759798535 | 0 |
| Method-3 | Clusters Without Feature Selection | TRUE | 0.75 | 56 | 28 | 2.810338389 | 0 |
| Method-4 | Clusters With Feature Selection | TRUE | 0.75 | 20 | 24 | 2.759798535 | 0 |

## 7. Conclusion

In this paper we proposed an algorithm to find the boundary points of each and every cluster by using the threshold valuesthen generate the new cluster for identified boundary points after thatcalculate the cluster quality using DB method. Identified proposed cluster quality is better than the existing cluster quality. At the same time it produces the optimum quality of the cluster by generating stabilized clusters at certain threshold value.

## References


1. ramesh, c., jena, g., &rao, k. R. (2017). Distributed and progressive feature selection algorithm for high dimensional data: a map reduce approach. *Journal of theoretical & applied information technology*, *95*(24).
2. Ramesh, C. R., Rao, K. R., & Jena, G. (2018). Fuzzy Clustering Algorithm Efficient Implementation Using Centre of Centres. *International Journal of Intelligent Engineering and Systems*, *11*(5), 1-10.
3. Nirmal, K. R., &Satyanarayana, K. V. V. REDIC K–Prototype Clustering Algorithm for Mixed Data (Numerical and Categorical Data) International Journal of Recent Technology and Engineering, Vol. 7,no. 6,pp. 1-
4. Yijun Sun “Iterative RELIEF for Feature Weighting: Algorithms, Theories, and Applications” IEEE transactions on pattern analysis and machine intelligence, vol. 29, no. 6, june 2007.

5. K. Kira and L.A. Rendell, "A Practical Approach to Feature Selection," Proc. Ninth Int'l Conf. Machine Learning, pp. 249- 256,1992.
6. B.SekharBabu, P. Lakshmi Prasanna ,Customer Data Clustering using Density based algorithm" International Journal of Engineering & Technology, 7 (2.32) (2018) 35-38.
7. D. Koller and M. Sahami, "Toward Optimal Feature Selection," Proc. 13th Int'l Workshop Machine Learning (ICML '96), pp. 284-292,1996
8. ]T.G. Dietterich, "Machine Learning Research: Four Current Directions," AI Magazine, vol. 18, no. 4, pp. 97-136, 1997.
9. Y.Vijay Bhaskar Reddy, L.S.S Reddy,.S.S.N.Reddy"Cross Breed Clustering Algorithm for High Dimensional Data" International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-9 Issue-1, November 2019
10. P.LakshmiPrasanna, S.Manogni, P.Tejaswini ,K.Tanmay Kumar , K.Manasa "Document Classification Using KNN with Fuzzy Bags of Word Representation" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7, Issue-6S, March 2019
11. Nikhath, A. K., &Subrahmanyam, K. (2019). Feature selection, optimization and clustering strategies of text documents. *International Journal of Electrical & Computer Engineering (2088-8708)*, *9*(2).
12. Rajkumar, K. V., Yesubabu, A., &Subrahmanyam, K. (2019). Fuzzy clustering and Fuzzy C-Means partition cluster analysis and validation studies on a subset of CiteScore dataset. *International Journal of Electrical and Computer Engineering*, *9*(4), 2760.
13. Potharaju, S. P., &Sreedevi, M. (2019). Distributed feature selection (DFS) strategy for microarray gene expression data to improve the classification performance. *Clinical Epidemiology and Global Health*, *7*(2), 171-176.
14. Manikandan, A., Danapaquiame, N., Gayathri, R., Kodhai, E., &Amudhavel, J. (2018). A Novel Clustering Algorithm for Big Data: K-Means-Fuzzy C Means. *BIOSCIENCE BIOTECHNOLOGY RESEARCH COMMUNICATIONS*, *11*(1), 85-93.
15. Vallabhaneni, R. B., & Rajesh, V. (2018). Brain tumour detection using mean shift clustering and GLCM features with edge adaptive total variation denoising technique. *Alexandria engineering journal*, *57*(4), 2387-2392.
16. Potharaju, S. P., Sreedevi, M., &Amiripalli, S. S. (2019). An Ensemble Feature Selection Framework of Sonar Targets Using Symmetrical Uncertainty and Multi-Layer Perceptron (SU-MLP). In *Cognitive Informatics and Soft Computing* (pp. 247-256). Springer, Singapore.
17. Potharaju, S. P., &Sreedevi, M. (2018). Correlation Coefficient Based Feature Selection Framework Using Graph Construction. *Gazi University Journal of Science*, *31*(3).
18. Rao, K. R., & Josephine, B. M. (2018, October). Exploring the Impact of Optimal Clusters on Cluster Purity.In *2018 3rd International Conference on Communication and Electronics Systems (ICCES)* (pp. 754-757).IEEE.
19. Satapathy, S. K., Mishra, S., Mallick, P. K., Badiginchala, L., Gudur, R. R., &Guttha, S. C. (2019). Classification of Features for detecting Phishing Web Sites based on Machine Learning Techniques. *International Journal of Innovative Technology and Exploring Engineering, volume8 (8S2)*, 425-430.
20. Potharaju, S. P., &Sreedevi, M. (2018). A Novel Cluster of Quarter Feature Selection Based on Symmetrical Uncertainty. *Gazi University Journal of Science*, *31*(2).