FRNC-KNNOM: AN INMPROVED FUZZY BASED REVERSE NEIGHBOUR CLUSTERING WITH KNN CLASSIFICATION OUTLIER MAPPING ALGORITHM

Devi.B[1]
M.Phil Research Scholar, Department of Computer Science,
Kongunadu Arts and Science College,
Coimbatore, India.

Vani. G[2]
Associate professor, Department of Computer Science,
Kongunadu Arts and Science College,
Coimbatore, India.

Abstract: Data mining techniques are used mainly of the clustering and document categorization, for calculating data points into identity related groups such that the points that belong to the similar group are additional similar than the points belonging to different clusters. Detection of anomalies or outliers points is a very significant issue in a high dimensional unsupervised datasets. Clustering has been employed by numerous researchers to solve such problems and choosing amount of *K* clusters without information loss. These issues of the existing techniques are making them inappropriate for selecting an amount of clusters in real world dataset. To overcome this problem without requiring choosing *K* value based clustering, this paper presents an improved Fuzzy based reverse neighbour clustering with KNN classification outlier mapping (FRNC-KNNOM) algorithm that can be easily allocated number of clusters in dynamic manner in MATLAB R2013a simulation. The proposed method follows a modified Fuzzy clustering and KNN Classification algorithm with reverse neighbour outlier mapping to discover the distance between data points. Finally, the proposed FRNC-KNNOM algorithm performs the Reverse Neighbour search and clusters process in dynamic manner.

Keywords: Data Mining, RNN, Fuzzy, KNN, Classification, Outlier.

I. INTRODUCTION

Data Mining, "The removal of hidden outlier's information from large databases", is a controlling new technology with great possible to help corporation focus on the most relevant information in their data warehouses. Data mining tools forecast future trends and behaviors, allowing businesses to create proactive, knowledge-driven choices [1]. The automated, prospective analyses obtainable by data mining move outside the analyses of precedent events provided by display tools typical of decision support systems. Data mining tools can reply business quires that conventionally were too time consuming to resolve. They clean databases for unknown patterns, finding analytical information that proficient's may neglect because it lies exterior their expectations.

In many data mining applications, the main step is discovering outliers (anomalies) in a raw dataset. Outlier detection for data mining is generally based on distance, clustering and spatial methods [2]. This paper compacts with locating outliers in large, multidimensional datasets. This work deals with Fuzzy clustering with KNN Classification techniques are computed with recognizing anomalies. The fuzzy clustering algorithm partitions a dataset into an amount of clusters, and then the outcomes are used to discover out the outliers from each cluster, using any one of the outlier's detection methods. This algorithm is enhanced in three manners. The primary is by using a Euclidean distance metric. The second and third enhancements are brought forward by automating the process of estimating cluster value and initial seed selection using the enhanced clustering algorithm.

Many researchers have projected different techniques to attain clustering results. Beside with managing a very large dataset, a robust clustering technique must satisfy some requirements such as scalability, dealing different types of attributes, discovering clusters of arbitrary shape, large dimensionality, capability to compact with noise and outliers, interpretability and usability. With clustering, time complexity increases with dealing large number of dimensions and large set of data objects [3]. Also the effectiveness depends upon the definition of similarity (or dissimilarity) among objects. Along with this, the output of clustering can be interpreted in different ways.

The performance of a clustering algorithm may be affected by the selected value of *K*. Therefore, instead of using a single predefined *K*, a set of values might be adopted. It is important for the number of values considered to be reasonably large, to reflect the specific characteristics of the data sets. At the same time, the selected values have to be significantly smaller than

the number of objects in the data sets, which is the main motivation for performing data clustering [4].

This paper presents an improved Fuzzy based reverse neighbor clustering with KNN classification outlier mapping (FRNC-KNNOM) algorithm that leverages a parallel execution of the Neighbor clustering through dense observations.

The main contributions of this paper are as follows:

- To develop an effective clustering algorithm is to without choosing a proper *K* value to determine the amount of cluster results and their information loss function.
- The proposed FRNC-KNNOM clustering is to outlier detection and correction using KNN with fuzzy cluster outlier mapping algorithms presented in this work.

The rest of the paper is organized as follows: Literature Review is detailed in Sect. 2. In Sect. 3, Research methodology in Sect. 3, Experimental results are described in Sect. 4; finally conclusion is in Sect. 5.

II. RELATED WORK

(*M. A. Cheema, X. Lin, W. Zhang, and Y. Zhang*, **2011**) [5] discussed a "given set of objects and a query *q*, a point *p* is called the reverse k nearest neighbor (RkNN) of *q* if *q* is one of the *k* closest objects of *p*. The authors introduced the concept of influence zone which is the area such that every point inside this area is the RkNN of *q* and every point outside this area is not the RkNN. The influence zone has several applications in location based services, marketing and decision support systems. It can also be used to efficiently process RkNN queries. First, they presented efficient algorithm to compute the influence zone. Then, based on the influence zone, we present efficient algorithms to process RkNN queries that significantly outperform existing best known techniques for both the snapshot and continuous RkNN queries. They also presented a detailed theoretical analysis to analyze the area of the influence zone and IO costs of our RkNN processing algorithms".

(*M. A. Cheema, W. Zhang, X. Lin, Y. Zhang, and X. Li*, **2012**) [6] authors studied the problem of continuous monitoring of reverse k nearest neighbors queries in Euclidean space as well as in spatial networks. Existing techniques are sensitive toward objects and queries movement. For example, the results of a query are to be recomputed whenever the query changes its location. They present a framework for "continuous reverse k nearest neighbor (RkNN) queries by assigning each object and query with a safe region such that the expensive recomputation is not required as long as the query and objects remain in their respective safe regions. This significantly improves the computation cost. As a byproduct, our framework also reduces the communication cost in client–server architectures because an object does not report its location to the server unless it leaves its safe region or the server sends a location update request".

(*Emrich, et.*, **2014**) [7] authors presented a Reverse nearest neighbor (RNN) queries in spatial and spatio-temporal databases have received significant attention in the database research community over the last decade. "A reverse nearest neighbor (RNN) query finds the objects having a given query object as its nearest neighbor. RNN queries find applications in data mining, marketing analysis, and decision making. Most previous research on RNN queries over trajectory databases assumes that the data are certain. In realistic scenarios, however, trajectories are inherently uncertain due to measurement errors or time-discredited sampling. They studied RNN queries in databases of uncertain trajectories. They proposed two types of RNN queries based on a well established model for uncertain spatial temporal data based on stochastic processes, namely the Markov model".

(*Toole,, et.*, **2015**) [8] authors discussed a flexible, modular, and computationally efficient software system to fill this gap. Their system estimates multiple aspects of travel demand using call detail records (CDRs) from mobile phones in conjunction with open- and crowd sourced geospatial data, census records, and surveys. They take away together numerous existing and new algorithms to generate representative origin–destination matrices, route trips through road networks constructed using open and crowd-sourced data repositories, and perform analytics on the system's output.

(*S. Yang, M. A. Cheema, X. Lin, and W. Wang*, **2015**) [9] authors showed that the performance of these algorithms is significantly improved even when a small buffer (containing 100 pages) is used. Finally, in each of the existing studies, the proposed algorithm is mainly compared only with its predecessor assuming that it was the best algorithm at the time which is not necessarily true as shown in our experimental study. Motivated by these limitations, they presented a comprehensive experimental study that addresses these limitations and compares some of the most notable algorithms under a wide variety of settings.

(*G. Casanova, E. Englmeier, M. E. Houle, M. Nett, E. Schubert, and A. Zimek*, **2017**) [10] authors discussed about given a query object q, reverse k-nearest neighbor (RkNN)

search aims to locate those objects of the database that have q among their k-nearest neighbors. The proposed an approximation method for solving RkNN queries, where the pruning operations and termination tests are guided by a characterization of the intrinsic dimensionality of the data. The method can accommodate any index structure supporting incremental (forward) nearest-neighbor search for the generation and verification of candidates, while avoiding impractically-high preprocessing costs. They also provided experimental evidence that our method significantly outperforms its competitors in terms of the tradeoff between execution time and the quality of the approximation.

III. RESEARCH METHODOLOGY

In this paper, proposed an improved Fuzzy based reverse neighbor clustering with KNN classification outlier mapping (FRNC-KNNOM) algorithm that can be easily allocated number of clusters in dynamic manner in MATLAB R2013a simulation. In order to know the methodology of FRNC-KNNOM in Data pre-processing, Fuzzy clustering, Reverse Nearest Neighbor Search with KNN Classification algorithm is performed. The proposed work flow diagram is described in figure 1.

A. *DATA PREPROCESSING*

Data preprocessing or data cleaning polices can be effectively rooted in collecting learning algorithms. In this process of irrelevance filter removes irrelevant features using a modified features, which assigns relevance values to features by treating training samples as points in feature space. For each sample, it finds the nearest "hit" (another sample of the same class) and "miss" (a sample of a different class), and adjusts the significance value of each feature according to the square of the feature difference between the sample and the hit and miss. Irrelevance Filter feature selection methods evaluate attributes prior to the learning process, and without specific reference to the clustering algorithm that will be used to generate the final result. The filtered dataset may then be used by any clustering algorithms.

B. *FUZZY CLUSTERING*

The Fuzzy clustering is to discover the data points as cluster centroid has to the optimal membership Link for estimating the centroids, and typicality is used for improving the disagreeable effect of anomalies. The function is composed of two expressions:

- The first is the fuzzy logic function and uses a Euclidean distance exponent,
- The second is fuzzificaion weighting function exponent; but the two coefficients in the objective function are only used as exhibitor of membership link and typicality.

The fuzzy aggregation assigns data points to c partitions by using optimal memberships. Let $X = \{x_1, x_2, x_3 \ldots x_n\}$ denote a set of data points to be portioned into c clusters, where x_i ($i = 1, 2, 3 \ldots n$) is the data points. The objective function is to discover nonlinear relationships among the data, kernel (root) methods use embedding linking's that connectivity features of data to new feature spaces. The proposed technique Fuzzy based kernel mapping (FKM) algorithm is an iterative clustering technique that minimizes the objective function.

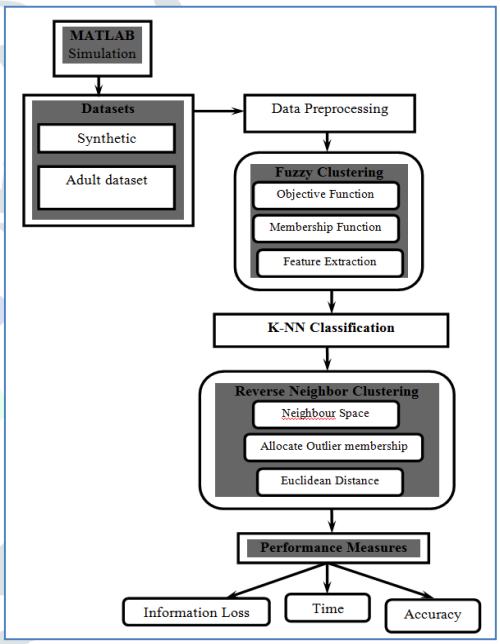

Fig. 1: Proposed flow diagram

Given an dataset, $X = \{x_1 \ldots x_n\} \subset Rp$, the original *KFCNC* algorithm partitions X into c fuzzy partitions by minimizing the following objective function as,

$$J(w,U,V) = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^{m} \| x_k - v_i \|^2 \quad \text{eqn. (1)}$$

Where c is the number of clusters and selected as a specified value, n the number of data points, u_{ik} the membership link of x_k in class i, satisfying the $\sum_{i=1}^{c} u_{ik} = 1$, m the quantity scheming clustering fuzzification, and V the set of cluster centers or prototypes ($v_i \in$ Rp).

C. *REVERSE NEAREST NEIGHBOR SEARCH WITH KNN CLASSIFICATION*

The proposed system implemented a new algorithm for outlier detection that has proven to be effective at detecting a variety of novel, interesting, and anomalous data behaviors. The new algorithm evaluates the local data distribution around a test data point and compares that distribution with the data distribution within the sample defined by its K nearest neighbors. The algorithm's success is based on the assumption that the distribution of distances between a true outlier and its nearest neighbors will be different from the distribution of distances among those neighbors by themselves. This assumption relies on the definition of an outlier as a point whose behavior (i.e., the point's location in parameter space) deviates in an unexpected way from the rest of the data distribution. Our algorithm quantifies this deviation, and uses that quantity as a measure of O(x), the "outlyingness" of the data point, or its "surprise index", or its "interestingness.

Algorithm 1: K-NN Classification with Reverse Neighbor Outlier Mapping in Adult dataset

Step 1: Read the Adult dataset features or attributes from the file $X = \{x_1, x_2, \ldots, x_n\}$

Step 2: Set the initial fuzzy clustering memberships from the dataset *X* into number of data clusters formation

Step 3: Set *K* to some value for nearest neighbor classification

Step 4: Normalize the attributes values in to double precision

Step 5: Find the candidate feature weighting element using distance method

Step 6: Sorting the feature weight in to ascending order

Step 7: For each feature finding the outlier distance to the point in candidate *k* nearest neighbors

- Determine the weight of each member of outlier candidate feature
- Search feature weighting element on to outlier points that are active in the cluster
- To map the outlier points according to the nearest distance clusters.
- Calculate the Execution Time has Elapsed Time = (End time – Start Time) / 1000
- Calculate the Information Loss as Loss = [(length(outlier), Correct Classification rate, number of training samples, number of classes, Cross validation)]
- Calculate the accuracy as Accuracy = (#average dissimilarity – lowest average dissimilarity)/max(cluster similarity)

The mathematical model for KNN classification with fuzzy cluster mapping algorithm shows local prior probabilities for outlier mapping process. Let X_i be an input sample with *p* features ($X_{i1}, X_{i2,\ldots,}X_{in}$) where X_i = (i = 1,2, …, n). The equation (1) find the similarity function among users is defined as follows:

$$d(x_i,x_o)=\sqrt{(X_{i1}-X_{o1})^2+(X_{i2}-X_{o2})^2+\cdots+(X_{in}-X_{op})^2} \quad \text{eqn. (2)}$$

where X_i is the input cluster feature and X_{op} is the outlier feature respectively represents the n^{th} attribute value of Adult dataset.

where m_i is a nearest neighbor to *X* if the distance

$$d(m_i, X) = \min_j\{d(m_j, X)\} \quad \text{eqn. (3)}$$

IV. RESULT AND DISCUSSIONS

The research work results describe a preliminary experimental evaluation of the outlier detection and correction using KNN with fuzzy cluster outlier mapping algorithms presented in this work. We implemented the proposed algorithms in MATLAB and compared them against a few other prominent outlier detection techniques. The proposed system, performed all the experiments on a Windows machine with a Intel core I5 processor with 3.20 Ghz speed and 8 GB of RAM. In this experiment study, we compare the performance of information loss, Computational Time and Accuracy of the Adult dataset.

Table 1 describes the performance of the Information loss metrics. The information loss is analyzed in terms of outlier detection error ratio results in figure 4.1. Information loss is defined as,

$$Loss = D^2 - O\frac{1}{1+\frac{N_{DC}-1}{N_{DC}}}\left(\frac{D}{2},\frac{N_{DC}-D}{2}\right) \quad \text{eqn. (4)}$$

where,

D: Dimensionality of the vector (2,3,4,5,...), O: OuTlier
NDc: The number of training samples per class (>D+1)

Table 1: Comparisons of Information Loss during number of runs

Methods	1	2	3	4	5
Existing KNN	0.75	0.72	0.6	0.55	0.25
Proposed FRNC-KNNOM	**0.56**	**0.42**	**0.35**	**0.27**	**0.17**

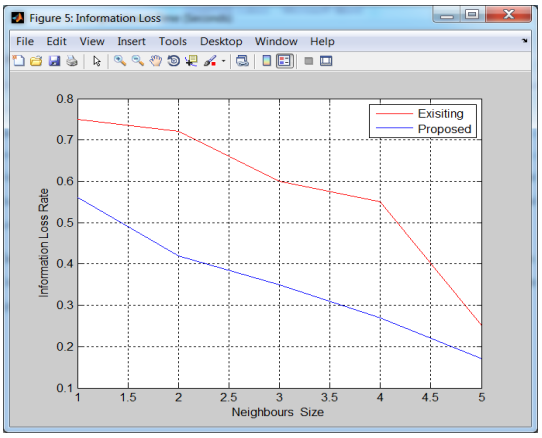

Fig. 2: Information Loss

Table 2 describes the performance of the system using metrics such as Time metrics. Computational time metrics is analyzed in terms of Outlier detection and correction in figure 4.2. Computational Time is defined as

$$\text{Computational Time} = \frac{Proces\ End\ time}{Process\ Start\ time} \text{ eqn. (5)}$$

Table 2: Comparisons of Computational Time (CPU seconds) during number of runs

Methods	1	2	3	4	5
Existing KNN	26	27	27	28	29
Proposed FRNC-KNNOM	**15**	**17**	**18**	**20**	**23**

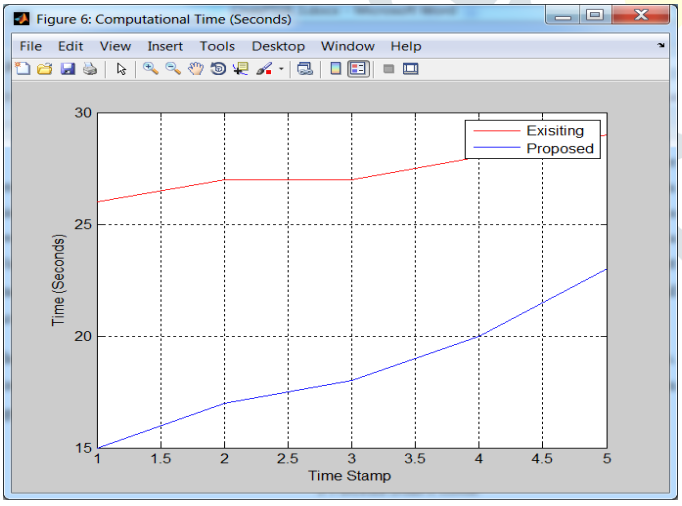

Fig. 3: Computational Time

Table 3 describes the performance of the system using metrics such as Accuracy metrics in figure 4.3. Accuracy is defined as,

$$Accuracy(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}} \text{ eqn. (6)}$$

where, a(i) is the cluster similarity, b(i) be the lowest average dissimilarity of *i* to any other cluster, of which *i* is not a member. The cluster with this lowest average dissimilarity is said to be the "neighboring cluster" of *i* because it is the next best fit cluster for point *i*.

Table 3: Comparisons of Clustering Accuracy

Methods	1	2	3	4	5
Existing KNN	86	88	89	92	93
Proposed FRNC-KNNOM	89	99	99.5	99.8	100

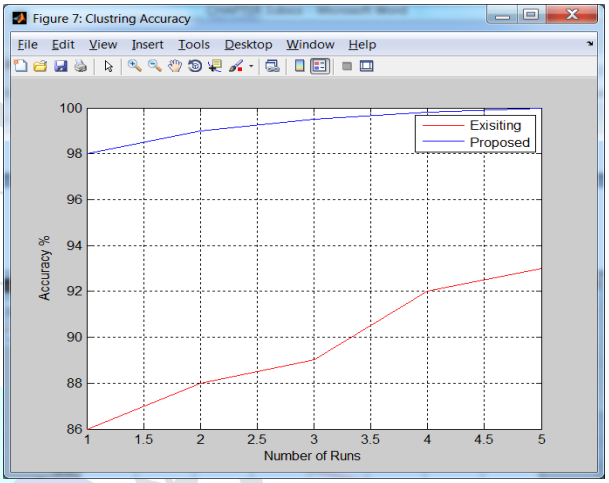

Fig. 4: Cluster Accuracy

V. CONCLSUION

In this paper, presents an improved Fuzzy based reverse neighbor clustering with KNN classification outlier mapping (FRNC-KNNOM) algorithm that can be easily allocated number of clusters in dynamic manner in MATLAB R2013a simulation. The proposed model to predict the neighbor cluster density using Reverse Neighbor KNN Classification with base of Fuzzy Clustering algorithm is applied to the several artificial and real-world datasets. Finally, the FRNC-KNNOM algorithm updates the amount of clusters, Neighbor similarity through Euclidean distance manner. In addition, proposed algorithm develops two processes, namely Fuzzy cluster and Reverse KNN outlier mapping algorithm, to further accelerate the algorithm. To perform extensive experiments to evaluate the proposed algorithm and the results demonstrate the efficiency of our algorithm.

REFERENCES

[1] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, The kdd process for extracting useful knowledge from volumes of data. Commun. ACM, 39(11):27-34, 1996.

[2] C. C. Aggarwal and C. K. Reddy, editors. Data Clustering: Algorithms and Applications. CRC Press, 2014.

[3] W. Dong, C. Moses, and K. Li, "Efficient k-nearest neighbor graph construction for generic similarity measures," in Proc. 20th Int. Conf. World Wide Web, 2011, pp. 577–586.

[4] Y. Lv, et al., "An efficient and scalable density-based clustering algorithm for datasets with complex structures," Neurocomput., vol. 171, no. C, pp. 9–22, Jan. 2016

[5] M. A. Cheema, X. Lin, W. Zhang, and Y. Zhang, "Influence zone: Efficiently processing reverse k nearest neighbors queries," in Proc. IEEE 27th Int. Conf. Data Eng., 2011, pp. 577–588.

[6] M. A. Cheema, W. Zhang, X. Lin, Y. Zhang, and X. Li, "Continuous reverse k nearest neighbors queries in Euclidean space and in spatial networks ," VLDB J., vol. 21, no. 1, pp. 69–95, 2012.

[7] T. Emrich, H. P. Kriegel, N. Mamoulis, J. Niedermayer, M. Renz, and A. Z€ufle, "Reverse-nearest neighbor queries on uncertain moving object trajectories," in Proc. Int. Conf. Database Syst. Advanced Appl., 2014, pp. 92–107.

[8] J. L. Toole, S. Colak, B. Sturt, L. P. Alexander, A. Evsukoff, and M. C. Gonzalez, "The path most traveled: Travel demand estimation using big data resources," Transp. Res. Part C: Emerging Technol., vol. 58, no. Part B, pp. 162–177, 2015.

[9] S. Yang, M. A. Cheema, X. Lin, and W. Wang, "Reverse k nearest neighbors query processing: Experiments and analysis," Proc. VLDB Endowment, vol. 8, no. 5, pp. 605–616, 2015.

[10] G. Casanova, E. Englmeier, M. E. Houle, M. Nett, E. Schubert, and A. Zimek, "Dimensional testing for reverse k-nearest neighbor search," Proc. VLDB Endowment, vol. 10, no. 7, pp. 769–780, 2017.