A Clustering Approach to Reduce Outliers in Large Data Sets

[1]Birinder Singh Sarao and [2]Kamaljeet Kaur
[1,2]Mata Gujri College, Fatehgarh Sahib, Punjab, India
[1]birsarao@gmail.com, [2]kkresearch18@gmail.com

Abstract Detection of outliers from large datasets is a complicated issue in data mining nowadays. Discovering outliers is very useful in the detection of unrevealed and unpredicted data in various areas like credit cards fraud detection, criminal behaviours, etc. Many outlier detection algorithms like depth-based outlier detection algorithm, K-Means, Distance-based, Iterative K-Means, etc. are already being used in various surveys, research and review articles. When used on small datasets, these algorithms performance is almost the same, but as the size increases from 50 documents to 1000, their efficiency starts varying. A new algorithm is proposed considering the bulkiness of data and complexity of time. Clustering of the data is done in a typical manner to optimize the process of reducing outliers. The proposed approach is an enhanced version of depth based clustering algorithm, which is implemented on real-world data sets and compared with other algorithms based on various parameters.

Index Terms: Big Data, K -Means, Clustering, Depth Based Outlier Detection, Outlier Detection, Cosine Similarity.

1. Introduction

The number of Internet users has increased by 80% in the last few years. Any nearby location like a hotel or restaurant or place of interest can be searched on the Internet. With this massive user base of Internet, the necessity to efficiently manage the data has also increased. Correct clustering of data and efficiently detecting outliers can yield fine search results [1]. In today's scenario, handling this vast data is referred to as Big Data [2].

1.1 Big Data

Big Data is also data, but it is enormous and is continuously growing exponentially [3, 4]. However, if this vast data is handled correctly, it can provide accurate results. Figure 1 shows the elements, which are wrongly placed, and their notation is shown below:

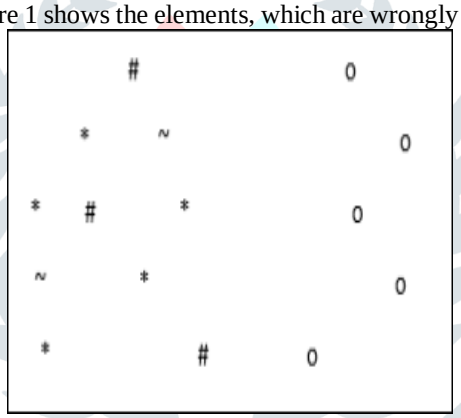

Figure1 Data Categories

Figure 1 contains different categories of data as detailed below:-

* = Restaurants
o = Hotels
~ = Hotels and Restaurants
# = Erroneous elements

In case a user wants to know about a restaurant and searches for it, then s/he might not get exact information on restaurants only but might get information on hotels and restaurants as well. Similarly, there are many practical examples where results may appear if data management is not proper [5].

2 Definition of Problem

In this research, the following problem to be solved is defined below:

$$P = findf(Dt, Cl, \alpha)$$

Where Dt represents data, Cl represents clusters and α represents the threshold of the term searched for the appropriate cluster. This research provides an ideal cluster solution in which the various elements get optimized, i.e., F-Measure, Recall and Precision.

Figure 2 described below represents explanation of false as well as true selection of data
Where,

Tp (True Positive)

Fp (False Positive)

Fn (False Negative)

◇Tn (True Negative)

/// is True data

If the selected data is not a true element, then it is false positive (Fp) and if the selected data is dissimilar to the searched one, then it will be considered as true positive (Tp). If the data is not appropriate and it is not shown then its true negative (Tn) otherwise, if the data element is not selected despite being true, then its false negative (Fn).

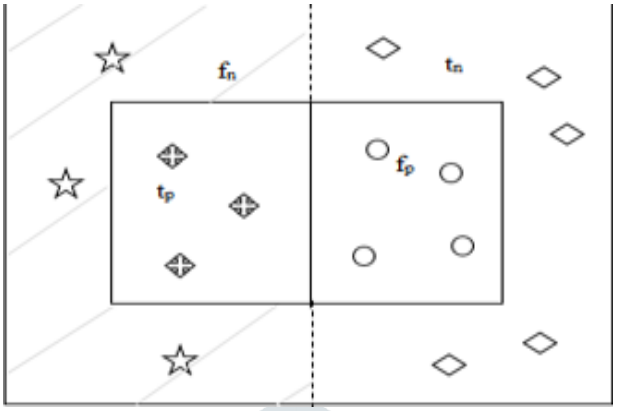

Figure2 Data Selection

$$\text{Precision} = \frac{\{\text{Docs selected}\} \cap \{\text{True Docs}\}}{|\{\text{Docs selected}\}|}$$

$$\text{Recall} = \frac{\{\text{TrueDocs}\} \cap \{\text{SelectedDocs}\}}{|\{\text{TrueDocs}\}|}$$

$$\text{F} - \text{Measure} = \frac{2 * \{\text{Precision} * \text{Recall}\}}{\text{Recall} + \text{Precision}}$$

3 Proposed Architecture

Existing algorithms of outlier detection or reduction work only with one kind of similarity index, which may lead to an uncertain behavior at the prediction of outliers. The proposed architecture is classified into further sections: -

1. Enhancing the documents with the similarity indexes [6].
2. Application of Depth-based outlier deduction on the basis of improved similarity indexes [7].
3. For early response, clustering would be done within the cluster using K-Mean [8, 9].

3.1 Datasets Used

The dataset used in the proposed algorithm is free and available easily on the Internet. These are text type of documents [10] as the similarity index has to be calculated. The data set contains a bulk text.
The dataset contains a large number of text files, around 5000 in the count. The average count of this set is 4800.

1. Dataset1 contains 4200 documents
2. Dataset2 contains 5100 documents
3. Dataset3 contains almost 4900 documents.

The proposed work has utilized three different types of datasets.

- Dataset1 contains www-pages collected from the computer department of different universities by worldwide knowledge base project of a particular group. In this 8,282 pages are categorized into six main classes [11].
- Dataset2 consists of courses, which are designed for the graduate students to give them a thorough grounding in methodologies, mathematics, algorithm and technology. Full information about the instructors, class timing, textbook, grading system, the home assignment is given [12].
- Dataset3 describes the lexical and semantic relation between Hindi words [13, 14].

3.2 Similarity Index

It shows the relationship between the two considered documents [6]. It indicates the closeness (the quality of being only a short distance away or apart in space or time) of the two documents with each other [15]. Various researchers have worked on various similarity indexes like the cosine similarity index, Gaussian similarity index, etc. The algorithm proposed in this research work covers a complicated pattern of similarity indexes, which is hybrid and combination of Gaussian and cosine similarity [3, 16].

3.3 Cosine Similarity(CS)

It is used to measure the similarity of different documents irrespective of their sizes [17].

Let us assume two document elements d1, d2

$$d1.d2 = |d1||d2|cos\theta$$

3.4 Gaussian Similarity

It is defined as the distance of Gaussian space between two documents considered above [18].

$$GS = gauss(d1.length) - gauss(d2.length)$$

3.5 Enhanced Similarity Index Algorithm(ESIA)

Enhanced Similarity Index is considered using the summation procedure on both Gaussian similarity and cosine similarity. The algorithm is presented below:

Enhanced Similarity Index Algorithm

Cosim= Function to calculate cosine similarity of documents
// Input is taken as docs and output is Cosim
Cosim=[];
SimCont= 0;
For l=0:length.docs
Curr_doc=docs(l);
For m= l+1:length.docs
K= {Cos (Curr_doc)-Cos (docs(m))};
Cosim[SimCont,0]=Curr_doc;
Cosim[SimCont,1]= docs(m);
Cosim[SimCont,2]= k;
SimCont= SimCont+1;
End for;
End for;
End function;

Gausim= It is a function for calculating the Gaussian Similarity Index of documents.
Gausim = [];
For l=0:length.docs
Curr_doc= docs(l);
Mu1=0;

$$Mu1 = Mu1 + \sum_{k=0}^{p} \frac{word\ value(k).Current\_doc}{p}$$

Mu2= 0;
For j= 1+ 1: docs

$$Mu2 = Mu2 + \sum_{k=0}^{p} \frac{wordvalue(k).docs(j)}{p}$$

Gausim(gauscount,0)= current_doc
Gausim(gauscount,1)= docs (m)
Gausim(gauscount,2)= Ratio of Mu1 and Mu2;
End for;
End function;;

ImproveSimilarity = It is used to calculate enhanced (Cosim,Gausim)
ImproveSim=[];
For l=0:length(Cosim)
ImproveSim(l,0)= Cosim(l,0);
ImproveSim(l,2)= Cosim(l,2)+Gausim(l,2);
End for;
End function;
The concept of calculating Radius is taken from Depth-based clustering algorithm [19].
Calculation of Similarity Index (SI) is as follows:

$$SI = \left(\sum_{k=1}^{x} doc_1 - doc_2 \right)$$

$$C = \sum_{k=1}^{x} \frac{doc_{kx}}{doc_{cnt}}$$

Algorithm Cosine Similarity Connection

```
If SI k1= found1
If doc.satisfy c1 (c,r)
c1(c1 count)= doc.Id;
 c0 count= c1 c0;
else
c2 (c2 count)= doc.Id;
c2 count= c2 count +1;
```

Radius calculation Algorithm

```
 [R1,R2,R3]= function to find radius (ImproveSim)
// Based on improved similarity, function returns 3 radius
// R1 is the max value of Improved Similarity Index
//R2 is the min value of Improved Similarity Index
Rt=  ImproveSim(0,3);
// To find max value
 For l=0: length (ImproveSim)
 If ImproveSim(l,3) >R1
 R1= ImproveSim(1,3);
end if;
end for;
// To find min value
  R2= ImproveSim(0,3)
  For l=0:length(ImproveSim)
  If ImproveSim(l,3)<R2
  End if
  End for
//To find the average of the radii calculated
   Sum= R1+R2
   R3= Sum/2;
//R3 is average of min & max radius values.
  End Function;
```

As shown in Table1, documents 1, 2 and 3 have relations with connecting document 4, respectively. If the positions of document 4 are changed in a row where document 2 is related to 4 and document 3 is related to 4 exists. So, the S.I (Similarity Index) in all the connections where 1 is related to 4, 2 is related to 4, and 3 is related to 4 are .76, .83 and .71, respectively. Out of these, .71 is the closest, which is for document 3 related to 4.

Table1: Connecting Documents

Improved Sim		
Main doc	Connecting Doc	I Sim
1	4*	.76
2	4*	.83
3	4*	.71
.	.	.
.	.	.
.	.	.
n-1	n*	SI

4 Simulation Results

Simulation work is done using various parameters which are described below [29, 30];

4.1 Precision

It is defined as the familiarity of two or more than two measurements with respect to each other. It can be written as:

$$Precision = \frac{Relevantdata - Retrieveddata}{Retrieveddata}$$

4.2 Recall

It is defined as the fraction of data, which is relevant and effectively retrieved from the existing documents [37]. It can be written as:

$$Recall = \frac{Relevantdata - Retrieveddata}{Relevantdata}$$

4.3 F-Measure

It is defined as the measure that sums up or combines the scores of precision and recall. It can be written as:

$$F - Measure = 2.\frac{Precision.recall}{Precision + Recall}$$

The proposed approach is compared with the traditional algorithms [21].The comparative results are presented in Figure 3.

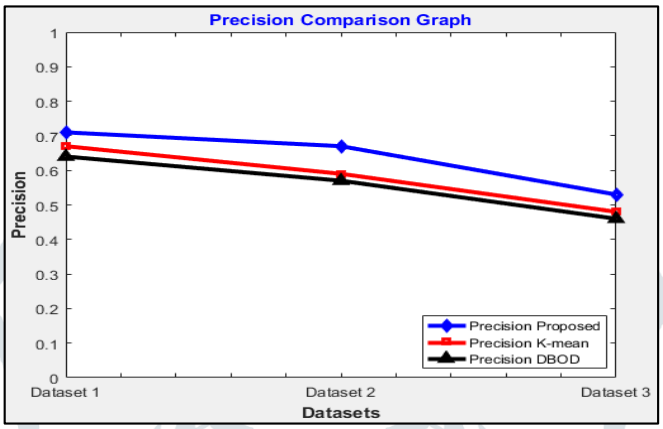

Figure3 Comparison of Precision

Table2: Comparison of Precision with different Algorithms

Datasets used	Precision of Proposed Approach	Precision of K-Mean	Precision of DBOD
Dataset 1	0.71	0.67	0.64
Dataset 2	0.67	0.59	0.57
Dataset 3	0.53	0.48	0.46

Figure 3 and Table 2 is showing the comparison of precision values with different datasets on different algorithms. It can be seen in Figure 3, that the precision of proposed approach is better than the traditional algorithms.

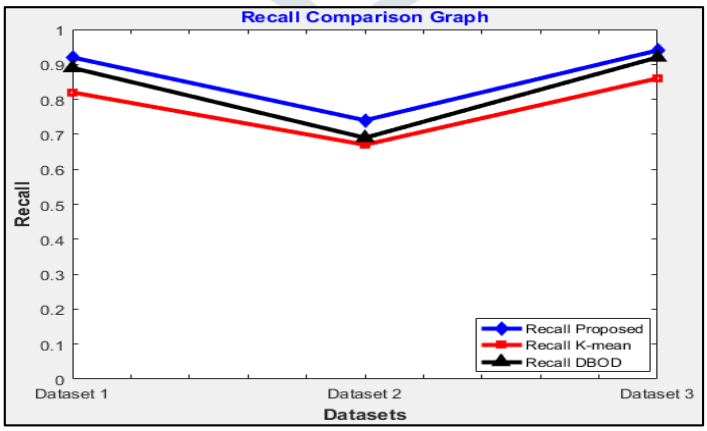

Figure4 Comparison of Recall

Table3: Comparison of Recall with different Algorithms

Datasets used	Recall of Proposed Approach	Recall of K-Means	Recall of DBOD
Dataset 1	0.92	0.82	0.89
Dataset 2	0.74	0.67	0.69
Dataset 3	0.94	0.86	0.92

Table 3 and Figure 4 show the evaluation of parameter recall with different datasets on different algorithms. The recall of ESIA is much better as compared to K-Mean and DBOD. In similar way, other following parameters are also evaluated and the results of them are shown below:

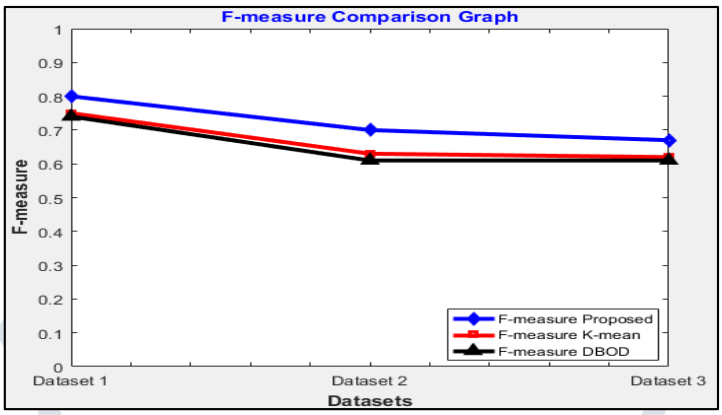

Figure5 Comparison of F-Measure

Figure 5 shows the comparison of parameter F-Measure with different existing algorithms and proposed algorithm on different datasets. Table 4 represents that F-Measure of the ESIA is far better than other existing algorithms.

Table4: Comparison of F-measure on different datasets

Datasets used	F-Measure of Proposed Approach	F-Measure of K-Mean	F-Measure DBOD
Dataset 1	0.80	0.73	0.74
Dataset 2	0.70	0.63	0.61
Dataset 3	0.67	0.62	0.61

Figure 6 depicts the total outlier percentage tested on 3 datasets. The clustering of proposed approach is done in such a manner that every document gets a second chance to be adjusted in some cluster and hence the chances of being a document to be an outlier are minimal.

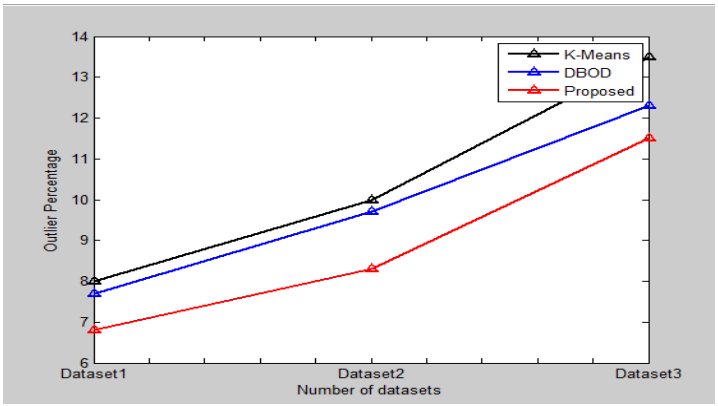

Figure6 Outlier Vs Datasets

5 Conclusion

Using a combined perspective of cluster identification and outlier detection, the ability to detect outliers is improved. Contrary to the traditional methods of clustering, the proposed algorithm provided more efficient outlier detection of large datasets. The objective of proposing the ESIA algorithm is to efficiently detect the outliers while filtering the data in the clustering process. Various parameters like Recall, Precision, F-measure were evaluated in the simulation. The detection of outliers by the ESIA algorithm was more efficient than the other existing algorithms. In future, this work can also be implemented on some real life application by considering some more parameters.

References

1. Jiang D, Tang C, Zhang A. Cluster analysis for gene expression data: a survey. IEEE Transactions on knowledge and data engineering. 2004 Oct 4;16(11):1370-86.
2. Chen, C. P., & Zhang, C. Y., Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. Information Sciences, 2014, 275: 314-347.
3. Hui Li, Di Wu, Gao-Xiang Li, Yi-Hao Ke, Wen-Jie Liu, Yuan-Huan Zheng, Xiao-La Lin. Enhancing Telco Service Quality with Big Data Enabled Churn Analysis: Infrastructure, Model, and Deployment[J]., 2015, 30(6): 1201-1214.
4. Lee, J., Lapira, E., Bagheri, B. & Kao, H. A., Recent advances and trends in predictive manufacturing systems in big data environment. Manufacturing Letters, 2013,1(1): 38-41 (2013).
5. Lin, A. D., Graydon, P. J., Busch, J. E., Caudill, M., Chinchor, N. A., Tseng, J. C. M. & Tijerino, Y. A., U.S. Patent No. 6,675,159. Washington, DC: U.S. Patent and Trademark Office. (2004).
6. Sawa, T. & Ohno-Machado, L., A neural network-based similarity index for clustering DNA microarray data. Computers in Biology and Medicine,2003, 33(1): 1-15
7. Stojmenovic, I., Russell, M. & Vukojevic, B. Depth first search and location based localized routing and QoS routing in wireless networks. Proceedings of 2000 in International Conference on Parallel Processing (ICPP 2000), 2000, 173-180, IEEE.
8. Chen, C. W., Luo, J. & Parker, K. J., Image segmentation via adaptive K-mean clustering and knowledge-based morphological operations with biomedical applications. IEEE transactions on image processing,1998: 7(12), 1673-1683.
9. Ahmad, A. & Dey, L., A k-mean clustering algorithm for mixed numeric and categorical data. Data & Knowledge Engineering, 2007,63(2):503-527.
10. Goyal, P., Mehala, N., Bhatia, D. & Goyal, N., Topical document clustering: two-stage post processing technique. International Journal of Data Mining, Modelling and Management, 2018, 10(2):127-170.
11. http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/
12. http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/mlc/
13. Guha, S., Meyerson, A., Mishra, N., Motwani, R., & O' Callaghan, L., Clustering data streams: Theory and practice. IEEE transactions on knowledge and data engineering, 2003, 15(3):515-528.
14. http://www.hindiwords.net
15. Liang, J., Shi, Q. & Zhao, X., Multi-view data ensemble clustering: a cluster-level perspective. International Journal of Machine Intelligence and Sensory Signal Processing, 2018,(2): 165-188
16. Fan Jianhua; Li Deyi;. An Overview of Data Mining and Knowledge Discovery[J].,Journal of Computer Science and Technology, 1998,13(4): 348-368
17. Muflikhah, L. & Baharudin, B., Document clustering using concept space and cosine similarity measurement. International Conference on Computer Technology and Development (ICCTD'09), 2009, Vol. 1, 58-62, IEEE
18. Wassermann, D., Bloy, L., Kanterakis, E., Verma, R. & Deriche, R., Unsupervised white matter fiber clustering and tract probability map generation: Applications of a Gaussian process framework for white matter fibers. NeuroImage, 2010,51(1): 228-241
19. Kriegel, H. P., Kröger, P., & Zimek, A., Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Transactions on Knowledge Discovery from Data (TKDD), 2009, 3(1):1.
20. Hu, R., Dou, W. & Liu, J., ClubCF: a clustering-based collaborative filtering approach for big data Application. IEEE transactions on emerging topics in computing, 2013, 2(3): 302-313.
21. Kumar, D., Bezdek, J. C., Palaniswami, M., Rajasegarar, S., Leckie, C. & Havens, T. C., A hybrid approach to clustering in big data. IEEE transactions on cybernetics,2016, 46 (10): 2372-2385.
22. Von Luxburg, U., A tutorial on spectral clustering. Statistics and computing, 2007, 17(4):395-416.
23. Ferragina, P. & Gulli, A., A personalized search engine based on Web-snippet hierarchical clustering. Software: Practice and Experience, 2008:38(2).
24. Brun, A., Knutsson, H., Park, H. J., Shenton, M. E. & Westin, C. F., Clustering fibre traces using normalized cuts. International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, Berlin, Heidelberg, 2004,368-375
25. Bicego, M., Murino, V. & Figueiredo, M. A., Similarity-based clustering of sequences using hidden Markov models. International Workshop on Machine Learning and Data Mining in Pattern Recognition, Springer, Berlin, Heidelberg, 2003, 86-95.
26. Ding, C. & He, X., Cluster merging and splitting in hierarchical clustering algorithms. Proceedings of IEEE International Conference on Data Mining (ICDM 2003), 2003, 139-146.
27. Bradley, P. S. & Fayyad, U. M., Refining Initial Points for K-Means Clustering. In ICML, july 1998, Vol. 98: 91-99.
28. Gandhi, B. S. & Deshpande, L. A., The survey on approaches to efficient clustering and classification analysis of big data. International Conference on Computing Communication Control and automation (ICCUBEA 2016), IEEE, 2016, 1-4.
29. Ma, Y., Yang, X., Wang, C. & Chen, F., An improved clustering algorithm and its application. International Journal of Wireless and Mobile Computing, 2017, 12(4): 358-363.
30. Kaur, K. & Garg, A., Performance Evaluation of Outlier-Detection Algorithms using various Parameters. International Journal of Applied Research on Information Technology, 2017, 8(2):141-151.