An Enhanced Similarity Index Approach to optimize clusters in Data

[1]Birinder Singh Sarao
1Mata Gujri College, Fatehgarh Sahib, Punjab, India

Abstract
Data Clustering being a serious issue as it may lead to major flaws in data sets. As clustering between various documents is based on the similarity index between the data files. This research paper uses cosine similarity and Gaussian similarity to calculate the radius of the clusters and then the performance of the same is analysed with the other existing algorithms like K-Means, DBOD etc based on various parameters like efficiencies, accuracy etc. Proposed algorithm is an enhanced version of the improved similarity indexes on the depth based clustering algorithm, which is implemented on real data sets. 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.

Index Terms: Clustering, K-Means, Dice-Coefficient, Gaussian Similarity, Outliers

1. Introduction

Clustering algorithms are one of the types of unsupervised machine learning. Unsupervised learning uses unlabelled data. Most of the clustering algorithms work on the distance calculation evaluation. Details are given as:-

a. Calculation by Distance Measures

The distance similarity measure calculates the distance between two or more data elements in a list. The distance can only be calculated between the vector values and hence the data is converted into numeral vectors. There are lot of conversion methods like word to vector model, ASCII conversion, and intermediate word value solutions [1, 2, 18].

i. Cosine Similarity: It is the cosine of the two vectors values.

$$Cos_{Sim} = \frac{A.B}{||A||\,||B||} \qquad \text{--------- (1)}$$

ii. Soft Cosine: It is soft distance measure in which the data element is dependent upon the second vector value. It is a modified form of Cosine Similarity

iii. Dice-Coefficient: It is a measure of distance which is based on common elements in two given document vectors [3].

$$Dice_{coefficient} = 2 \times \frac{A \cap B}{|A||B|} \qquad \text{----------- (2)}$$

2. Proposed Architecture

The main objective of the algorithm is to calculate the similarities between documents. Cosine and Gaussian similarity mutually develops improved similarity algorithm. After combining both similarities, resultant similarity is computed**.**

Clustering of an algorithm is divided into two sections as follows:
1) In this section, find radius function covers the evaluated radius [3].
2) This section covers the radius for creating cluster [4, 5, 6].

Initially two clusters are created by the proposed algorithm whose algorithmic formation is presented below.

Cluster Formation of Algorithm

```
[clust1,clust2]=function to create initial clusters(R1,R2,R3)Improvesim
Clust1=[]; At first both clusters would be vacant
Clust2=[];
Clust1count= 0;
Clust2count= 0;
// At initial stage, first element is considered in first cluster
Clust1[clust1count=ImproveSim (0,1);
Clustcount=clustcount + 1;
R= connection numbers in enhanced similarity
For i=0: R
Present_elem= ImproveSim(i,2) // 1st connection collection
altrpositions=[]; // holds the alternating position of element existing presently
altrpositions=find{ImproveSim(i:2)==Present_elem}
altrsimvalues=improveSim(altrpositions, 3);
```

```
kp=findminimum (Altrvalues)
ks=Altrposition(ks);
If kp> R3
Clust1[Clust1count]=ImproveSim(ks,2)
```

2.1 Graphical User Interface (GUI) of Similarity Index Calculation

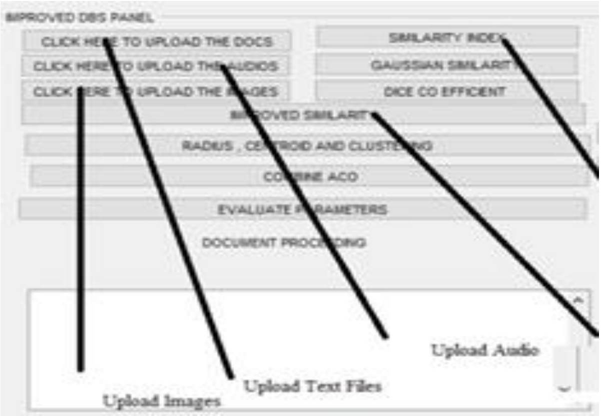

Figure1 GUI Interface

Figure 1 shows the processing of GUI (Graphical User Interface). In this figure, "Click here to upload the Audio" in this button we upload the Audio data. We have used only Audio data for this research.

3. Tool Utilized

In this research work, different algorithms are implemented on a same platform using MATLAB R2016a.The parametric based results and some details about the tool used in the work are discussed below:

Table1 Hardware and Software Detail

Computer	Core 2 Duo or superior
Random Access Memory	64 bit
Platform	Windows 7
Another Hardware	Mouse and Keyboard
Software	Mat lab (Matrix laboratory)

The hardware as well as software required to simulate the entire process is listed in Table1.The full name for MATLAB is Matrix Laboratory, which basically presents the LINPACK and EISPACK (package system project) (linear system packages) established by Matrix Software. Matrix Laboratory is a language of high-performance technical computing. It has a variety of visualization, programming environment and computing capabilities.

MATLAB is the latest programming language, it also has a sophisticated data structure, including built-in tools for debugging, editing and help with object-oriented programming. These MATLAB capabilities make it an educational and excellent search tool. The unified application is collected in a package called the Toolbox[11,12].

To implement research work, different categories of datasets are used. These datasets are used in all the attained objectives of the research work. The description of one category of data sets is described below:

3.1 Description of Audio Data

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) have 7356 total number of files (total size: 24.8 GB). The storage contains 24 professional actors (12 females, 12 male), vocalizing two lexically-matched statements in a neutral North American accent. Speech includes calm, happy, sad, angry, fearful, surprise, and disgust expressions, and song contains calm, happy, sad, angry, and fearful emotions. Every expression generated two different level of emotional intensity (normal, strong), with an additional neutral expression. All conditions are present in different modality formats: Audio-only (16bit, 48 kHz .wav), Audio-Video (720p H.264, AAC 48 kHz, .mp4), and Video-only (no sound). Note, there are no song files for Actor_18.

Audio-only files:
Audio-only files of all actors (01-24) are present in two different zip files (~200 MB each):

- Speech file (Audio_Speech_Actors_01-24.zip, 215 MB) have 1440 files: 60 trials per actor x 24 actors = 1440.
- Song file (Audio_Song_Actors_01-24.zip, 198 MB) contains 1012 files: 44 trials per actor x 23 actors = 1012.

4. Simulation Results

4.1 Efficiency

Efficiency is calculated with required average execution time to complete execution of an algorithm [7,8].

$$Efficiency\ =\ (ClusteredDocs * 2\ /\ TotalDocs) * 100$$

4.2 Accuracy

It is the proximity of a computation to the true value which is calculated by taking true positive and true negative with a fraction of true positive, true negative and false positive with false negative [7,9,10].

$$Accuracy = \frac{Tp + Tn}{Tp + Tn + Fp + Fn}$$

$\mathrm{WhereTp = Truepositive,\ \ Tn = Truenegative, Fp = falsepositiveand\ Fp = falsenegative}$

Table 2 shows the evaluation of parameter Efficiency and Accuracy with datasets on different algorithms. It represents that accuracy and efficiency of the proposed algorithm is far better than other existing algorithms.

Table2: Comparative Analysis on different datasets based on various parameters

Datasets used	Efficiency of Proposed Approach	Accuracy of Proposed Approach	Efficiency of K-Mean	Accuracy of K-Mean	Efficiency of DBOD	Accuracy of DBOD
Dataset 1	169.64	68.35	103.35	55.13	100.15	45.21
Dataset 2	182.21	82.26	113.13	61.63	102.19	47.31
Dataset 3	194.73	91.02	124.52	64.24	103.42	49.10

]

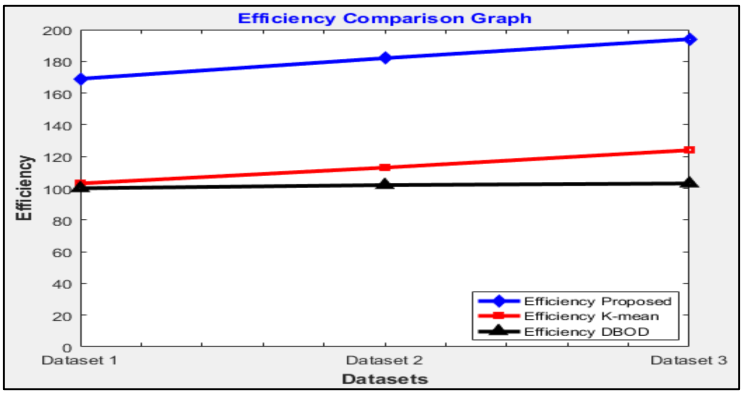

Figure2 Comparison of Efficiencies

Figure 2 shows the comparison of parameter Efficiency with different algorithms including proposed algorithm on audio dataset. It depicts that proposed approach is more efficient as compared to others.

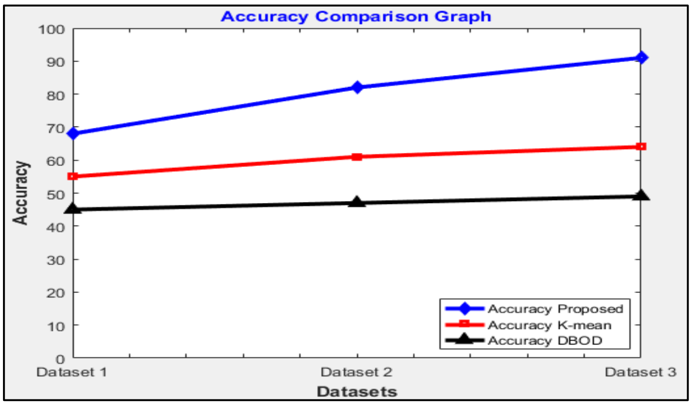

Figure3 Comparison of Accuracies

Figure 3 shows that accuracy of proposed algorithm is better than DBOD and K-Means algorithms [15].The proposed algorithm holds an average accuracy of 89% whereas DBOD and K-Means hold 45% and 56%, respectively.

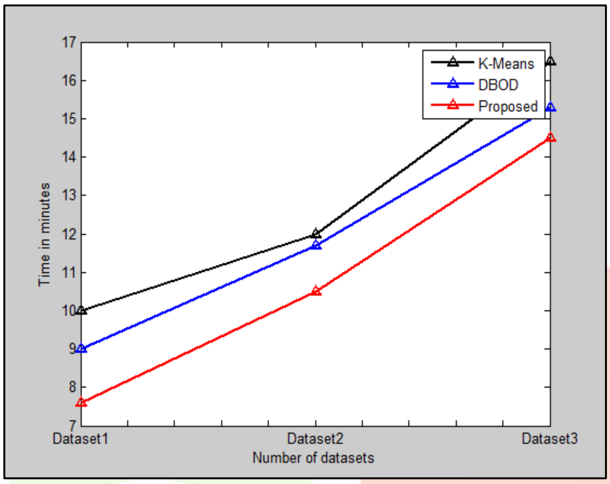

Figure4 Time Consumption in Minutes

Figure 4 represents the total time (minutes) consumed in order to cover all the text files. It depicts that on an average the total time consumed by the proposed approach is about 10 minutes i.e the time taken by DBOD (11.5 minutes) and K Means (12.9 minutes). Thus the time efficiency of the proposed algorithm is 15 % better than the other algorithms.

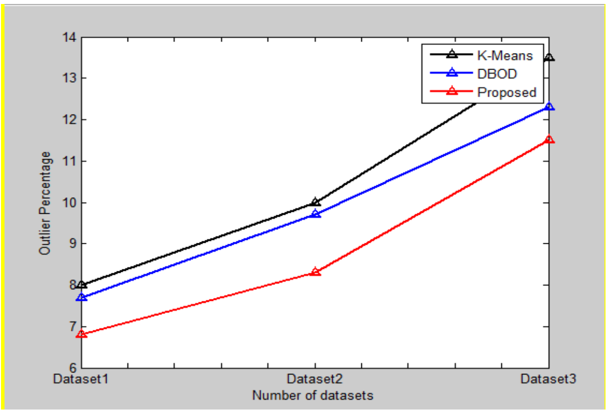

Figure5 Outlier Vs Datasets

Figure 5 depicts the total outlier percentage tested on datasets after the formation of clusters. 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.

5. Conclusion

The objective of the proposed algorithm is to efficiently form the clusters from the data sets and further which will result in efficiently detecting the outliers. Various parameters like accuracy, efficiency, time consumed and detection rate of outliers based on the cluster formation is computed.

Current research work has a lot future responsibilities. The optimization in the current research can be achieved by the Swarm Intelligence Category algorithm's like Ant Colony Optimization. Deep learning can be taken as an option in machine learning

References

1. Saini, A., Minocha, J., Ubriani, J. & Sharma, D. (2016, April). New approach for clustering of big data: DisK means. International Conference in Computing, Communication and Automation (ICCCA), pp.122-126, IEEE (2016)
2. 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.
3. Pandove, D. & Goel, S., A comprehensive study on clustering approaches for big data mining. In Electronics and Communication Systems (ICECS), 2015 2nd International Conference on Electronics and Communication Systems (ICECS), pp. 1333-1338, IEEE (2015).
4. Liu C, White M, Newell G. Detecting outliers in species distribution data. Journal of biogeography. Jan; 45(1):164-76(2018).
5. Garg A, Kaur K. An Efficient Method to Detect Outliers in High Dimensional Data. Journal of Computational and Theoretical Nanoscience. Sep 1; 16(9):pp. 3938-44 (2019).
6. Garg A, Kaur K. An Evolutionary Architecture for High Dimensional Data Optimization to Remove Data Redundancy. Journal of Advanced Research in Dynamical and Control Systems. July; 11(5):pp 3938-44 (2019).
7. Fan C, Xiao F, Li Z, Wang J. Unsupervised data analytics in mining big building operational data for energy efficiency enhancement: A review. Energy and Buildings. Jan 15; 159:296-308 (2018).
8. Wang, J., Du, P., Hao, Y., Ma, X., Niu, T. and Yang, W., An innovative hybrid model based on outlier detection and correction algorithm and heuristic intelligent optimization algorithm for daily air quality index forecasting. Journal of environmental management, 255, p.109855.(2020)
9. Kamalov, F., & Leung, H. H.Outlier detection in high dimensional data. Journal of Information & Knowledge Management, 2040013.(2020)
10. Singh DS, Singh G., "Big Data: A Review", International Research Journal of Engineering and Technology (IRJET)2017 april,(4): 822-824
11. Srivastava S. Weka: a tool for data preprocessing, classification, ensemble, clustering and association rule mining. International Journal of Computer Applications.1;88(2014).
12. Rodríguez-Mazahua L, Rodríguez-Enríquez CA, Sánchez-Cervantes JL, Cervantes J, García-Alcaraz JL, Alor-Hernández G. A general perspective of Big Data: applications, tools, challenges and trends. The Journal of Supercomputing. 1;72(8):3073-113.(2016)
13. Richa G, Sunny G and Anuradha S. Big Data: Overview., IJCTT, Vol 9, Number 5,(2014)
14. Naik N, Jenkins P, Savage N, Katos V. Big data security analysis approach using computational intelligence techniques in R for desktop users. IEEE Symposium Series on Computational Intelligence (SSCI) 2016 Dec 6 (pp. 1-8). (2016)
15. Chaudhari N, Srivastava S. Big data security issues and challenges. In2016 International Conference on Computing, Communication and Automation (ICCCA) 2016 Apr 29 (pp. 60-64).

16. Narang SK, Kumar S, Verma V. Knowledge discovery from massive data streams. In Web semantics for textual and visual information retrieval (pp. 109-143). IGI Global.(2017)
17. VikramPudi, PRadha Krishna. “Data Mining”, Oxford University Press, First Edition, 2009
18. Han J, Pei J, Kamber M. Data mining: concepts and techniques. Elsevier; 2011 Jun 9.