Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 10, October 2014, pg.62 – 69

RESEARCH ARTICLE

HANDLING REAL TIME DATA SETS USING STREAM MINING TECHNIQUES

Ms. S.Ranjitha Kumari[1], Dr. P.Krishna Kumari[2], S.Shylaja[3]

[1]Department of Computer Applications (MCA), Bharathiar University, India

[2]Department of Computer Applications (MCA), Bharathiar University, India

[3]Department of Computer Science, India

[1] ranjithakumari@rvsgroup.com; [2] kkumari@rvsgroup.com; [3] shylurose.ss@gmail.com

Abstract—

The Clustering is one of the most important techniques in data mining. It aims partitioning the data into groups of similar objects. That is refered to as clusters. This research compares the StreamKM++ algorithm with the existing work, such as AP, IAPKM and IAPNA. The StreamKM++ algorithm is a new clustering algorithm from the data stream and into constructs a good clustering of the stream, using a small amount of memory and time. Many researchers have done their work with static clustering algorithm, but in real time the data is dynamic in nature. Such as blogs, web pages, audio and video, etc., hence, the conventional static technique doesn't support in real time environment. In this work, the StreamKM++ algorithm is used which achieves high clustering performance over traditional AP, IAPKM and IAPNA. The experimental result shows StreamKM++ algorithm achieves the best result compared with existing work. It has increased the average accuracy rate and reduced the computational time, and memory.

Keywords— Data mining, clustering, clustering algorithm, StreamKM++, Data sets

I. INTRODUCTION

The Data mining is the process of discovering useful information (i.e. patterns) underlying the data. Powerful techniques are needed to extract patterns from large data because traditional statistical tools are not efficient enough anymore. Clustering is an important data mining technique that puts together similar objects into a collection in which the objects exhibit certain degree of similarities. Clustering also separates dissimilar objects into different groups. This has made clustering an important research topic of diverse fields such as pattern recognition, bioinformatics and data mining. It has been applied in many fields of study, from ancient Greek astronomy to present-day insurance industry and medical. Astronomy to present-day insurance industry and medical. In Existing Research, Affinity Propagation (AP) clustering has been successfully used in a lot of clustering problems. However, most of the applications deal with static data. This paper considers how to apply AP in incremental clustering problems. Firstly, it point out the difficulties of Incremental Affinity Propagation (IAP) clustering, and then propose two strategies to solve them. Correspondingly, two IAP clustering algorithms are proposed. They are IAP clustering based on K-Medoids (IAPKM) and IAP clustering based on Nearest Neighbor Assignment (IAPNA). Five popular labeled data sets, real world time series and a video are used to test the performance of IAPKM and IAPNA. Traditional AP clustering is also implemented to provide benchmark performance. Experimental results show that IAPKM and IAPNA can achieve comparable clustering performance with traditional AP clustering on all the data sets. Meanwhile, the time cost is dramatically reduced in IAPKM and IAPNA. Both the effectiveness and the efficiency make IAPKM and IAPNA able to be well used

in incremental clustering tasks. Affinity Propagation is a clustering algorithm that identifies a set of exemplar points that are representative of all the points in the data set. The exemplars emerge as messages are passed between data points, with each point assigned to an exemplar. AP attempts to find the exemplar set which maximizes the net similarity, or the overall sum of similarities between all exemplars and their data points.

II. CLUSTERING

Clustering is the task of segmenting a heterogeneous population into a number of more homogeneous sub groups or clusters. Clustering and clusters are not synonymous. A clustering is an entire collection of clusters; a cluster on the other hand is just one part of the entire picture. Clustering is a division of data into group of similar objects. Each group, called cluster consist of objects that are similar amongst themselves and dissimilar compared to objects of other groups. [8]

III.CLUSTER ANALYSIS

The process of grouping a set of physical or abstract object into classes of similar objects is called clustering. A cluster is a collection of data object that are similar to one another within the same cluster and are dissimilar to the object in other clusters.

Cluster analysis is an important human activity. One learns how to distinguish between cats and dogs, or between animals or plants, by continuously improving subconscious clustering schemes. Cluster analysis has been widely used in numerous applications, that including pattern recognition, data analysis, image processing, and market research by clustering.[6]

IV. CLUSTERING ALGORITHM

4.1 StreamKM++ ALGORITHM:

The StreamKM++ is a new k-means clustering algorithm for data streams. It computes a small weighted sample of the data stream and solves the k-means problem on this sample using the k-means-++ algorithm. We use two new techniques. First, we use non-uniform sampling similar to the k-means++ algorithm. This leads to a rather easy implementable algorithm and to a runtime which has only a low dependency on the dimensionality of the data. Second, we develop a new data structure called coreset tree in order to significantly speed up the time necessary for sampling non-uniformly during the coreset construction.

We are able to describe our clustering algorithm for data streams. To this end, let m be a fixed size parameter. First, we extract a small coreset of size m from the data stream by using the merge-and-reduce technique. Every time when two samples representing the same number of input points exist we take the union (merge) and create a new sample (reduce).

4.1.1 K-means++Algorithm

Algorithm k-MEANS (P, k)

1 choose initial cluster centers $c_1 \ldots c_k$ uniformly at random P

2 **repeat**

3 partition P into k subsets $P_1 \ldots P_k$, such that P_i, $1 \leq i \leq k$,

Contains all points whose nearest center is c_i

4 replace the current set of centers by a new set of centers $c_1 \ldots c_k$,
such that center c_i, $1 \leq i \leq k$, is the center of gravity of Pi

5 **until** the set of centers has not changed.

4.1.2 Adaptive Seeding Algorithm

Algorithm Adaptive Seeding (P, k)

1 choose an initial center c_1 uniformly at random from P

2 $C \leftarrow \{c_1\}$

3 **for** i ← 2 **to** k

4 choose the next center c_i at random from P, where the probability

of each p ε P is given by D^2 (p,C) / cost(P,C)

5 C ← C ∪ {ci}

V. COLLECTING THE DATA

In this paper, there are popular Six labeled data sets are used to compare the proposed algorithm with the existing algorithm. Iris Data set and Wine Data set This is the data sets which are taken from the http://www.ics.uci.edu/~mlearn/MLRepository.html UCI Repository.
The Six of the most popular Data Sets are used to evaluate the clustering algorithms. In data set Car and Yeast, the distribution of categories is seriously imbalance. However, it's not the focus of the paper, so only parts of the three data sets are used. In data set Car, four categories of objects are used, and each category consists of 65 objects. Data set Yeast contains 10 categories, where the most four and the data set of NSL KDD data set is using for dynamic environment and this is consider the 42 Attributes and Instance of 25600 objects are used. Each category contains 163 objects. Each data set is divided into six parts. The first part is used as initial objects, and the left objects are added in five times. More details can be found in Table 2. e.g. Iris, traditional AP clustering is implemented on the first100 objects, and the left objects are added 10 by 10. When new objects are arriving.

5.1.1 Data Description:

Data set	No of Objects	No of Attribute	Usage of Data set
Iris	150	4	Whole
Wine	178	13	Whole
Car	569	30	Whole
Yeast	1728	6	Partly
WDBC	1484	8	Partly
KDD Cup	25600	42	Partly

In this table use Six labeled data sets to evaluate the proposed algorithm. Six of the most popular Data Sets in are used to evaluate the clustering algorithms. A brief description is given in Table 1.

VI. RESULT OVER PROPOSED ALGORITHM WITH SIX DATA SETS

6.1 Performance Evaluation:

The performance of the proposed Work is evaluated with the existing approaches. It is analyzed with the proposed scheme in terms of Average Accuracy, Computational time, Memory Usage and Number of Iterations. Experimental results show that StreamKM++ Algorithm is comparable to previous work which requires Five Labeled data sets for improving the Accuracy and reducing the Time, memory in clustering.

6.1.1 Accuracy of Proposed Algorithm (StreamKM++):

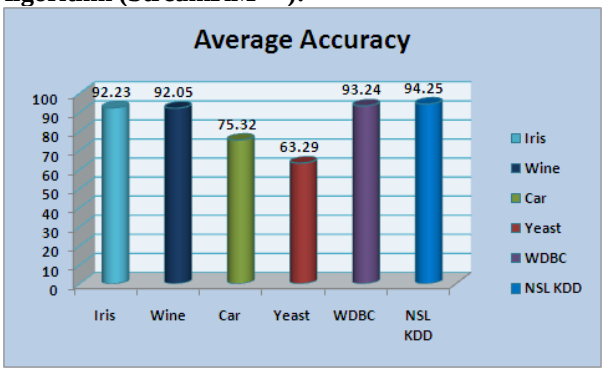

Figure 6.1.1 :Accuracy Values of Six Labeled Data set

This graph shows the Accuracy of StreamKM++clustering algorithm for most relevant to the popular Six Data sets. The Data sets in X-axis and the Accuracy value percentage in the Y-axis are measured.

6.1.2 Computational Time of Proposed Algorithm (StreamKM++) :

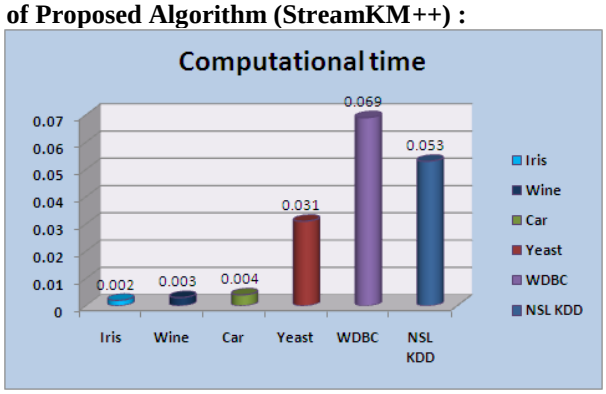

Figure 6.1.2: Computational Time of Six Labeled Data set

This graph shows the Computational Time of StreamKM++clustering algorithm for most relevant to the popular five Data sets. The Data sets in X-axis and the Computational Time in Seconds in the Y-axis are measured.

6.1.3 Memory Usage in Proposed Algorithm (StreamKM++) :

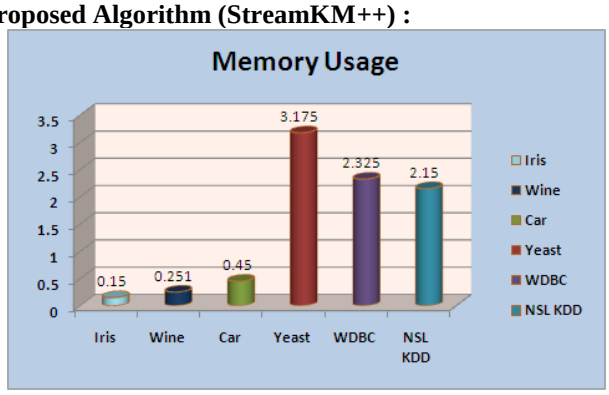

Figure 6.1.3: Memory Used in the Six Labeled Data set

This graph shows the Memory Usage in StreamKM++clustering algorithm for most relevant to the popular five Data sets. The Data sets in X-axis and the Memory Usage in MB in the Y-axis are measured.

VII. IMPLEMENTATION OF THE PROPOED WORK

7.1 RESULTS

The Following Implementation of Algorithms are used Popular six labeled Data sets. They are explained in details within the table and graph.

7.1.1 AVERAGE ACCURACY:

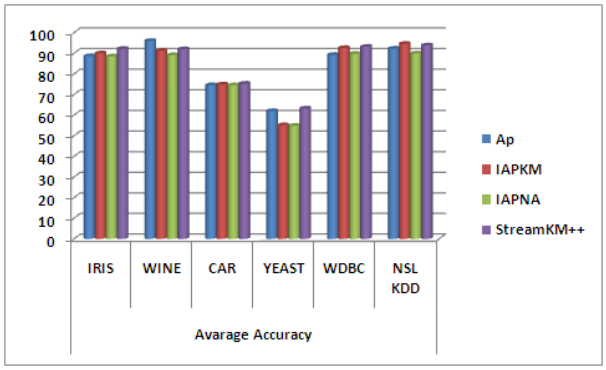

Figure 7.1.1 The Average Accuracy of AP, IAPKM, IAPNA and StreamKM++ Algorithms

This graph shows the Accuracy or measurement of the system between the Affinity Propagation, IAPKM, IAPNA and StreamKM++ algorithm for most relevant to the Six Labeled Data sets. The Data sets in X-axis and the Accuracy value in percentage in the Y-axis and right side the Algorithms are measured. The Accuracy value of StreamKM++ is higher than the other three algorithms.

Table 7.1.1: The Average Accuracy of AP, IAPKM, IAPNA and StreamKM++ Algorithms

S.No	Algorithm	Data sets	Accuracy
1	Affinity Propagation	Iris	88.67%
		Wine	96.07%
		Car	74.61%
		Yeast	62.04%
		WDBC	89.29%
		NSL KDD	92.34 %
2	IAPKM	Iris	90.04%
		Wine	91.25%
		Car	74.95%
		Yeast	55.19%
		WDBC	92.65%
		NSL KDD	94.67%
3	IAPNA	Iris	88.54%
		Wine	89.19%
		Car	74.56%
		Yeast	54.87%
		WDBC	89.82%
		NSL KDD	89.99%
4	StreamKM++	Iris	92.23%
		Wine	92.05%
		Car	75.32%
		Yeast	63.29%
		WDBC	93.24%
		NSL KDD	93.89%

7.1.2 COMPUTATIONAL TIME IN (SECONDS):

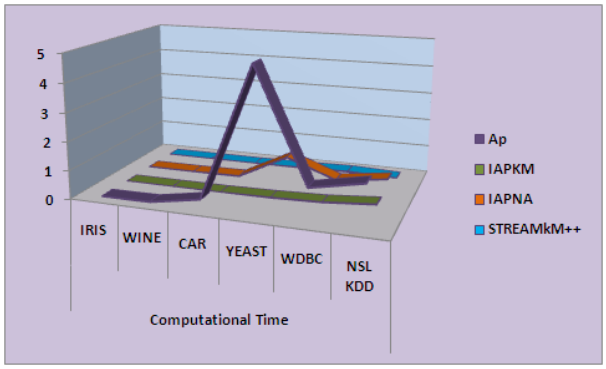

Figure 7.1.2 The Computational Time of AP, IAPKM, IAPNA and StreamKM++ Algorithms

This graph shows the Accuracy or measurement of the system between the Affinity Propagation, IAPKM, IAPNA and StreamKM++ algorithm for most relevant to the Six Labeled Data sets. The Data sets in X-axis and the Computational Time in Seconds in the Y-axis and right side the Algorithms are measured. The Computational Time of StreamKM++ is Slightly better than the other three algorithms.

Table 7.1.2: The Computational Times of AP, IAPKM, IAPNA and StreamKM++ Algorithms

S.No	Algorithm	Data sets	Computational Time
1	Affinity Propagation	Iris	0.025 Sec
		Wine	0.041 Sec
		Car	0.347 Sec
		Yeast	4.972 Sec
		WDBC	1.173 Sec
		NSL KDD	1.523 Sec
2	IAPKM	Iris	0.002 Sec
		Wine	0.002 Sec
		Car	0.003 Sec
		Yeast	0.032 Sec
		WDBC	0.087Sec
		NSL KDD	0.234 Sec
3	IAPNA	Iris	0.015 Sec
		Wine	0.017 Sec
		Car	0.045 Sec
		Yeast	0.946 Sec
		WDBC	0.295 Sec
		NSL KDD	0.531 Sec
4	StreamKM++	Iris	0.002 Sec
		Wine	0.003 Sec
		Car	0.004 Sec
		Yeast	0.031 Sec
		WDBC	0.069 Sec
		NSL KDD	0.007 Sec

7.1.3 MEMORY USAGE IN (MB):

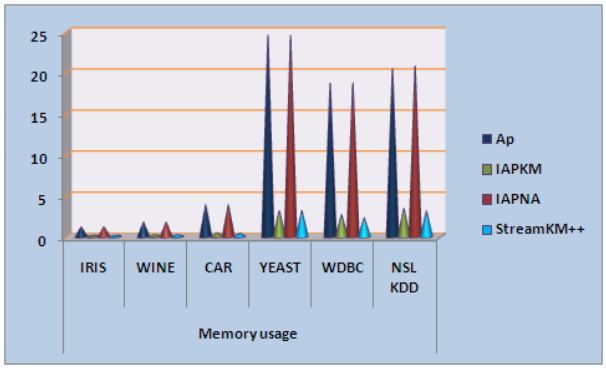

Figure 7.1.3: The Memory usage of AP, IAPKM, IAPNA and StreamKM++ Algorithms

This graph shows the memory or measurement of the system between the Affinity Propagation, IAPKM, IAPNA and StreamKM++ algorithm for most relevant to the Six Labeled Data sets. The Data sets in X-axis and the Memory Usage in MB in the Y-axis and right side the Algorithms are measured. The memory usage of StreamKM++ is reduced than the other three algorithms.

S.No	Algorithm	Data sets	Memory Usage
1	Affinity Propagation	Iris	1.215 MB
		Wine	1.768 MB
		Car	3.855 MB
		Yeast	24.50 MB
		WDBC	18.69 MB
		NSL KDD	20.45 MB
2	IAPKM	Iris	0.167 MB
		Wine	0.271 MB
		Car	0.520 MB
		Yeast	3.185 MB
		WDBC	2.697 MB
		NSL KDD	3.456 MB
3	IAPNA	Iris	1.215 MB
		Wine	1.769 MB
		Car	3.855 MB
		Yeast	24.50 MB
		WDBC	18.69 MB
		NSL KDD	20.765 MB
4	StreamKM++	Iris	0.150 MB
		Wine	0.251 MB
		Car	0.450 MB
		Yeast	3.175 MB
		WDBC	2.325MB
		NSL KDD	3.124 MB

VIII. CONCLUSION AND FUTURE WORK

The Data mining process is to extract useful information from the large database. And it involves the outlier detection, classification, clustering, summarization and regression. The clustering algorithm is one of the most important technique in Data mining. It aims partitioning the data into groups of similar objects. That is refered to as cluster. And the many researchers have done their work with clustering algorithm in static data. But, in real time the data is dynamic in nature such as blogs, web pages, video surveillance, etc.

Hence, the conventional static technique doesn't support in real time environment. This research compares the StreamKM++ algorithm with the existing work such as AP, IAPKM and IAPNA.

The experimental result shows StreamKM++ algorithm achieves the best result compared with existing work. It has increased the average accuracy and reduced computational time, and memory.

REFERENCES

[1] [Arun. K. Pujari] "Data Mining Techniques", Universities, press (India) Limited 2001, ISBN81- 7371-3804.

[2] [J. Han, M. Kamber, and J. Pei], "Data Mining: Concepts and Techniques", 3rd edition, Morgan Kaufmann, 2011. p. 444.S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," *IEEE Electron Device Lett.*, vol. 20, pp. 569–571, Nov. 1999.

[3] [T.W. Liao] "Clustering of Time Series Data: A SurveyPatternRecognition", vol. 38, no. 11, pp. 1857-1874, Nov. 2005

[4] [A.K. Jain] "Data Clustering: 50 Years Beyond K-means," PatternRecognition Letters, vol. 31, no. 8, pp. 651-666, June 2009.

[5] [S. Guha, A. Meyerson, N. Mishra, R. Motwani, and L. OCallaghan], "Clustering Data Streams: Theory and Practice," IEEE Trans. Knowledge and Data Eng., vol. 15, no. 3, pp. 515-528, May 2003.

[6] [A. Likas, N. Vlassis, and J.J. Verbeek],"The Global k-means Clus-tering Algorithm," Pattern Recognition, vol. 36, no. 2, pp. 451-461, Feb. 2003. .

[7] [F.R. Kschischang, B.J. Frey, and H.A. Loeliger], "Factor Graphs and the Sum-product Algorithm," IEEE Trans. Information Theory, vol. 47, no. 2, pp. 498-519, Feb. 2001.

[8] [J.S. Yedidia, W.T. Freeman, and Y.Weiss], "Constructing Free-Energy Approximations and Generalized Belief Propagation Algo-rithms," IEEE Trans. Information Theory, vol. 51, no. 7, pp. 2282-2312, July 2005.

[9] [W. Hwang, Y. Cho, A. Zhang, and M. Ramanathan], "A Novel Functional Module Detection Algorithm for Protein-protein Inter-action Networks," Algorithms for Molecular Biology, vol. 1, no. 1, pp. 1-24, Dec. 2006.
[10] [X. Zhang, C. Furtlehner, and M. Sebag], "Frugal and Online Affinity Propagation," Proc. Conf. francophone sure l'Apprentissage (CAP '08), 2008.
[11] [L.Ott and F. Ramos], "Unsupervised Incremental Learning for Long-term Autonomy," Proc. 2012 IEEE Int. Conf. Robotics andAutomation (ICRA '12), pp. 4022-4029, May 2012,.
[12] [D. Chakrabarti, R. Kumar, and A. Tomkins],"Evolutionary Clus-tering," Proc. Knowledge Discovery and Data Mining (KDD '06), pp. 554-560, Aug. 2006.
[13] [M. Charikar, C. Chekuri, T. Feder, R. Motwani],"Incremental Clustering and Dynamic Information Retrieval," Proc. ACM Symp.Theory of Computing (STOC '97), pp. 626-635, 1997.

Authors Profile

Ms.S.Ranjitha kumari is an Assistant Professor in Rathnavel Subramaniam College of Arts and Science, at Affiliated to Bharathiar University. She has more than nine years of teaching experience. Her areas of interest are Network Security and Machine Learning. She has published 15 papers in the national and international journals.

Dr.P.Krishna kumari is working as Director in Dept. of Computer Science and Applications, Rathnavel Subramaniam College of Arts & Science, Coimbatore, India. She got her Doctoral Degree in Computer Science. She has got seventeen years of experience in Academics. She has published 11 Research Papers in International and National Journals, 14 Research Papers in International and National conferences.

S.Shylaja is now a M.Phil. Research Scholar in Rathnavel Subramaniam College of Arts and Science, at Affiliated to Bharathiar University. She is Received B.Sc (CS) degree in Angappa College of Arts and Science. Malumichampatti, Coimbatore in 2011, and Completed M.Sc (CS) in Government Arts and Science College, at Affiliated to Bharathiar University. Coimbatore in 2013. And her specialization is Data Mining.

