Dr. V.Ramesh, International Journal of Computer Science and Mobile Applications,
Vol.3 Issue. 5, May- 2015, pg. 1-8 ISSN: 2321-8363

The Distributed K-Means Clustering Over Peer-To-Peer Mesh Networks

Dr. V.Ramesh
Professor, Department of IT, Mahatma Gandhi Institute of Technology, Hyderabad, India
E-Mail: v2ramesh634@yahoo.co.in

Abstract

K means Clustering deals with the problem of wireless mesh networks in Peer-to-Peer environments with distributed data, computing nodes, and decentralized connections. Peer-to-peer computing is distributed computing standard to a large sequence of applications that involves exchange of information among a huge number of peers with little centralized coordination. It is monitoring clusters in the data residing at the different nodes of a Peer-To-Peer wireless mesh network. This assumes that all data are available at a single location. The data sources are distributed over a large-scale Peer-To-Peer wireless mesh network and are collected from a central location and then clustered. The solution takes a decentralized approach, where peers (nodes) only synchronize with their immediate topological neighbours in the underlying communication network. The algorithm is adapted to dynamic Peer-To-Peer wireless mesh network where existing nodes drop out and new nodes join in during the execution of the algorithm and the data in network changes.

Keywords: ***Distributed, peer-to-peer, wireless mesh networks, K-means clustering.***

1. Introduction

Wireless Peer-To-Peer Mesh Network (Wireless Mesh Networks P2P) is a promising new technology which is being adopted as the wireless internetworking solution for the near future. Characteristics of Peer-To-Peer Wireless mesh networks such as rapid deployment and self configuration make Wireless mesh networks suitable for transient on-demand network deployment scenarios such as disaster recovery, hard-to-wire buildings, conservative networks and friendly terrains. However, Peer-To-Peer wireless mesh networks are highly decentralized, dynamic and normally include thousands of nodes. Also, Peer-To-Peer wireless mesh network usually have routing assistance and the notion of clients or servers. This imposes several challenges for distributed clustering in Peer-To-Peer wireless mesh networks. First, it is not practical to have global synchronization in large-scale Peer-To-Peer wireless mesh networks. Also, there are frequent topology changes caused by frequent failure and recovery of peers. Finally, there are frequent on- the-fly data updates on each peer. The routers in the infrastructure backbone are static and have better power, computation and storage resources. In a hybrid mesh, there are several client mesh network. By associating each mesh client network with one router of infrastructure mesh, the management of the whole wireless mesh network would become simple. Each mesh client network can be managed by a boundary router. This router is responsible to provide addresses, routing assistance, mobility management, power management and network monitoring to the mesh client networks. Security mechanism can also be enhanced by centralizing the Peer-To-Peer mesh client network. With the implementation of this scheme, each mesh client network is now centrally managed by the manager router of that region. But the over all mesh network is still distributed. Each manager router communicates with other routers, collaborates and manages the whole wireless mesh network. In the conventional management scheme, each node is managed by the

centralized server. While in our proposed scheme, router nodes manage client mesh networks in a distributed way and then these routers are managed by a centralized manager. Peer-To-Peer Wireless Mess Networks are also an attractive technology for long-lived infrastructure network such as wireless municipal area network in dense metropolis, heterogeneous networks. If one node drops out of the network, due to hardware failure or any other reason, its neighbours simply find another route. Extra capacity can be installed by simply adding more nodes. Mesh networks may involve either fixed or mobile devices. The principle is simple: data will hop from one device to another until it reaches a given destination. One advantage is that, like a natural load balancing system, with the installation of more devices, more bandwidth becomes available. Since this wireless infrastructure has the potential to be much cheaper than the traditional networks, many wireless community network groups are already creating wireless mesh network Peer-To-Peer wireless mesh networks can be mainly categorized into three types according to their architecture: Infrastructure/Backbone wireless mesh networks, Client wireless mesh networks and Hybrid wireless mesh networks. Infrastructure/Backbone wireless mesh networks consist of Mesh routers which are relatively static: make up a backbone and provide an infrastructure for the clients. These routers are usually gateways to wired networks or the Internet. Client wireless mesh networks like conventional ad hoc networks consist of mobile wireless nodes. These are infrastructure-less networks with dynamically changing topology and mobility. In Client Peer-To-Peer wireless mesh networks every node needs to perform the task of self configuration and routing as there is no router available. A hybrid wireless mesh networks consists of many ad hoc components (mobile clients wireless mesh networks) and an infrastructure wireless mesh networks.

On the other hand, the backbone routers are relatively static in nature or have very limited mobility. Mesh routers have wider transmission ranges. Each ad hoc component is connected to one of the routers present in the router backbone. Each Router manages its own ad hoc component, providing addresses, routes to destination, authentication and secure communication to nodes present in its ad hoc region.

The small transmission range limits the number of neighbouring nodes, which in turn increases the frequency of topology change, owning to node mobility. We discuss the addressing, routing assistance, mobility management, power management, and network monitoring and security assistance by this mechanism

2. Related Work

Several papers have dealt with the issues of Distributed K-Means clustering over Peer-To-Peer mesh networks. Related literature can be grouped into two categories: Finding K-nearest neighbor and Distributed Clustering in wireless mesh Peer-To-Peer networks. The problem of analysing data which are scattered over a such huge and dynamic set of nodes, where each node is storing possibly very little data but where the total amount of data is immense due to the large number of nodes.

2.1 Clustering Distributed Data Streams in Peer-to-Peer Environments

This paper describes a technique for clustering homogeneously distributed data in peer-to-peer Environment like sensor networks. The proposed technique is based on principles of K-means algorithm. It works in a localized asynchronous manner by communicating with the neighbouring nodes. This offers extensive theoretical analysis of the algorithm that bounds the error in the distributed clustering process compared to the centralized approach that requires all the observed data to a single site [1].

2.2 Towards Data Mining in Large and Fully Distributed Peer-to-Peer Overlay networks

In this paper, the concept of distributed clustering using K points is explained to the extreme. This paper targets the problem of analyzing data which are scattered over a huge and dynamic set of nodes, where each node is storing possibly very little data but where the total amount of data is immense due to the large number of nodes [2].

2.3 Distributed Data Clustering can be Efficient and Exact

Data clustering is one of the basic techniques in scientific data analysis and data mining. It partitions a data set into

groups of similar items, as measured by some distance metric. The data set sizes have grown rapidly with the exponential growth of computer storage and increasingly automated business and manufacturing processes. To cluster such large and distributed data sets, efficient distributed algorithms are called for to reduce the communication overhead, central storage requirements, and computation time, as well as to bring the resources of multiple machines to bear on a given problem as the data set sizes scale-up[3].

2.4 Progressive Distributed Top-k Retrieval in Peer-to-Peer Networks

In this paper, the benefits of best match/top-k queries in the context of distributed peer-to-peer information infrastructures are discussed and shown to extend the limited query processing in current peer-to-peer networks by allowing the distributed processing of top-k queries, while maintaining a minimum of data traffic. Our algorithm is based on dynamically collected query statistics only, no continuous index update processes are necessary, allowing it to scale easily to large numbers of peers, as well as dynamic additions/deletion peers [4].

2.5 On Efficient Top-k Query Processing in Highly Distributed Environments

In this paper, the advances in centralized database management systems show a trend towards supporting rank-aware query operators, like top-*k*, that enable users to retrieve only the most interesting data objects[5]. A challenging problem is to support rank-aware queries in highly distributed environments.

2.6 Distributed Page Ranking in Structured P2P Networks

This paper discusses the techniques of performing distributed page ranking on top of structured peer-to-peer networks. Distributed page ranking are needed because the size of the web grows at a remarkable speed and centralized page ranking is not scalable. The relationship between convergence time and bandwidth consumed is also discussed [6].

2.7 Distributed Data Mining in Peer-to-Peer Networks

Distributed K means clustering deals with the problem of cluster analysis for environments such as distributed data, computing nodes and peers. Peer-to-peer computing is emerging as a new distributed computing paradigm for many novel applications that involve exchange of information among a large number of peers with little centralized coordination [7].

2.8 The Price of Validity in Dynamic Networks

In this paper, we propose an ensemble paradigm for distributed classification in P2P networks. Under this paradigm, each peer builds its local classifiers on the local data and the results from all local classifiers are then combined by plurality voting. To build local classifiers, we adopt the learning algorithm of *pasting* bites to generate multiple local classifiers on each peer based on the local data[8].

3. Distributed K-Means Clustering On Peer-To-Peer Mesh:

3.1 Problem overview:

K-Means clustering partitions a collection of data tuples into K disjoint, exhaustive groups (clusters), where K is a user-specified parameter. The goal is to find the clustering which minimizes the sum of the distances between each data tuple and the centroid of the cluster to which it is assigned. K-means starts with an initial set of randomly chosen K centroids and moves on iteratively.

3.2 K-Means algorithm

K is the sample mean of each cluster. $\mathbf{x}$ refers to each of our examples, and $\mathbf{b}$ contains our ″K-Means Clustering is an algorithm among several that attempts to find groups in the data. In pseudo code, it is shown to follow this procedure:

Initialize $\mathbf{m}_i$, $i = 1,\cdots,k$, for example, to k random $\mathbf{x}^t$

Repeat

For all $\mathbf{x}^t$ in X

$b_i^t \leftarrow 1$ if $\| \mathbf{x}^t - \mathbf{m}_i \| = \min_j \| \mathbf{x}^t - \mathbf{m}_j \|$

$b_i^t \leftarrow 0$ otherwise

For all $\mathbf{m}_i$, i = 1,⋯,k

$\mathbf{m}_i \leftarrow$ sum over t ($b_i^t \mathbf{x}^t$) / sum over t (b_i^t)

Until $\mathbf{m}_i$ converge

The vector $\mathbf{m}$ contains a reference estimated.

The algorithm covers the following steps.

1) Choose some manner in which to initialize the m_i to be the mean of each group (or cluster), and do it.
2) For each example in your set, assign it to the closest group (represented by m_i).
3) For each m_i, recalculate it based on the examples that are currently assigned to it.
4) Repeat steps 2-3 until m_i converge

Here is an example showing how the means $\mathbf{m}_1$ and $\mathbf{m}_2$ move into the centers of two clusters.

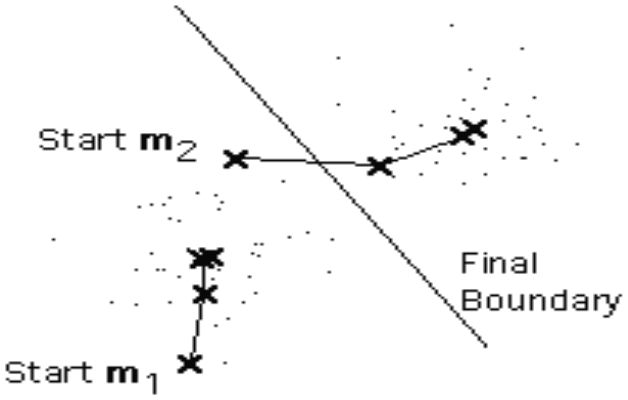


Fig. 1. The means $\mathbf{m}_1$ and $\mathbf{m}_2$ move into the centers of two clusters.

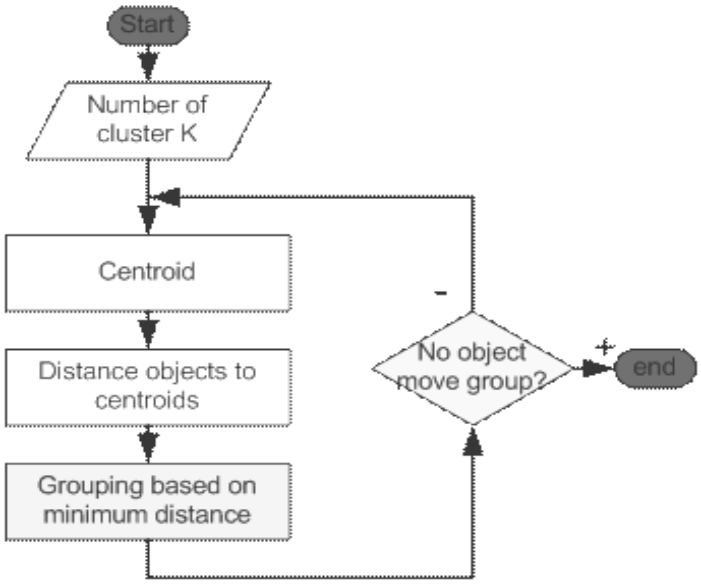


Fig 2.Flow chart for K means clustering algorithm

3.3 Implementation of K-Means algorithm.

In the new initialization method, the clustering algorithm will only be performed for several iterations during each run. After each run, initial points, which can be used to form the cluster with good structural similarity, are chosen and their distance is checked against that of all points already selected in the initialization array. If the minimum distance of new points is greater than the specified distance, these points will be added to the initialization array.

```
private static int findK(Sample s, double minSD)
{
    // Idea 2: split sample while standard deviation decreases
    double tolerance = 0.001, thisSD;

    s.sort();

    if (minSD < 0) minSD = s.getStandardDeviation();

    Sample workingSample1 = getHalf(1, s);
    Sample workingSample2 = getHalf(2, s);

    thisSD = workingSample1.getStandardDeviation();
    if (thisSD+tolerance < minSD)
        return 1 + findK(workingSample1, thisSD);

    thisSD = workingSample2.getStandardDeviation();
    if (thisSD+tolerance < minSD)
        return 1 + findK(workingSample2, thisSD);

    return 0;
}
```

Fig.3. Recursive function of FindK

The algorithm can easily adapt to dynamic Peer-To-Peer mesh network where existing nodes drop out and new nodes join in during the execution of the algorithm and the data in network changes. The solution also handles the Node failure and Topology changes.

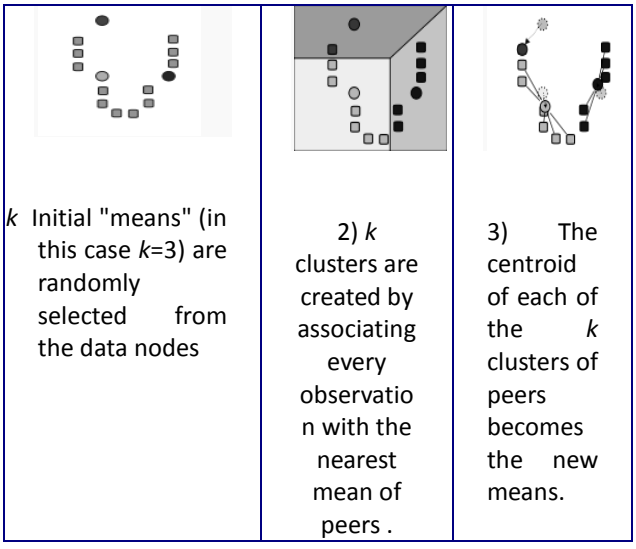


Fig .4. Demonstration of K-Means clustering algorithm

4. Clustering of wireless mesh in Peer-To-Peer environments: A distributed approach

4.1 Distributed Peer-to-Peer K-Means Clustering

This section describes the proposed P2P K-Means Clustering algorithm in a peer-to-peer mesh network. Using distributed and decentralized approach. Here first we describe K-Means algorithm.

K-Means clustering Method

A K-Partition of $X = \{x_1, x_2, x_3, \ldots, x_n\}$ can be conveniently represented by a $K \times n$ matrix known as partition matrix $U = [uik]$, $i = 1,2,\cdots,K$ $k=1,2,\cdots,n$ where uik is either 0 or 1, indicating that the pattern x_k belongs respectively cluster i.

K-Means [1,5,54] is a widely used technique for crisp partitioned clustering. The minimizing criterion used to characterize good clusters for K-Means partitions is defined as

$$J(U,V) = \sum_{i=1}^{k} \sum_{k=1}^{n} (uik) D_{ik}^2 (v_i, x_k) \quad (1)$$

Here U is a partition matrix ; $V=\{v_1, v_2, v_3, \ldots, v_n\}$ represents K cluster centers; $v_i \in IR^N$ and $D_{ik}(v_i,x_k)$ is the distance from x_k to v_i.

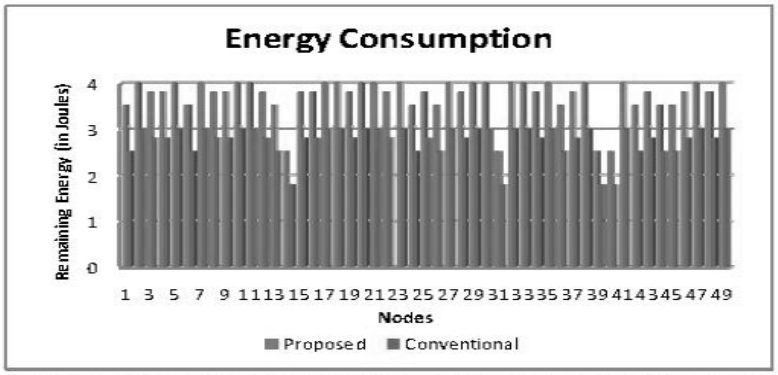
In the K-Means algorithms, the K initial seeds are first chosen randomly to represent the K centroids. Thereafter, the data points are assigned to the cluster of the closest centroid. This provides a partition matrix $U = [u_{ik}]$. After the assignment phase is over, the centroids are recomputed as follows:

$$v_i = \sum_{k=1}^{n} (u_{ik})\, x_k \,, \; i \le i \le K \quad \sum_{k=1}^{n} u_{ik}$$

A common strategy for generating the approximate solutions of the minimization of the problem in (1) is by iteratively performing the reassignment of the points to the closest centroids, and updating the centroids of the cluster with the mean of the points assigned to the same cluster.

4.2 Experimental results

The energy consumption by the nodes running with different with different management schemes. Each node is provided with 10 joules of initial energy. As the nodes perform transmission and receive messages their energy level is decreased. The graph in Fig. 4 shows that conventional centralized management scheme uses high amount of energy which means it has much higher amount of transmissions than our proposed scheme.



Shows the amount of energy remain at each client node in the WMN after 1000 seconds simulation.

4.3 Simulation Analysis

From the results, we observe that our proposed distributed management scheme consumes very little amount of energy, hence it has low overhead. In conventional management scheme each client node sends information to the centralized management server. This requires more messages and thus produces huge overhead in the network. In our distributed scheme, client nodes communicate with only the manager router of their mesh network. The rest of the communication is the responsibility of the router nodes. This reduces the overhead from the client nodes and consumes less energy.

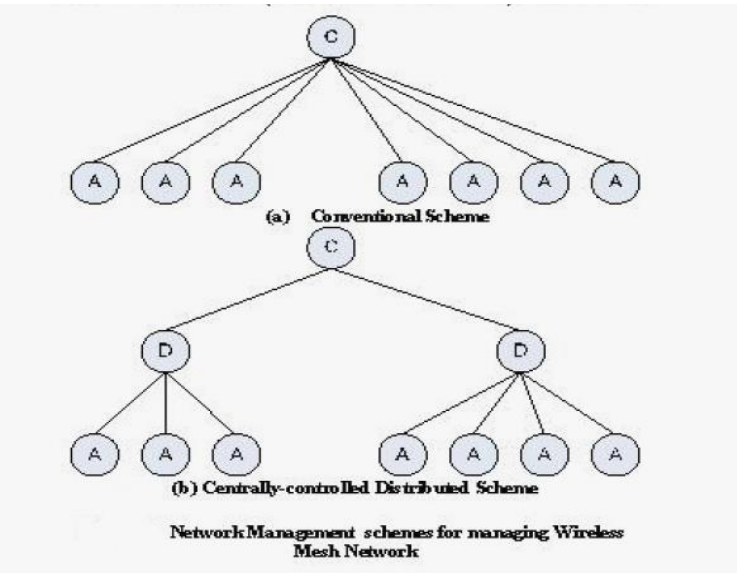


Fig. 6. Network management schemes (a) conventional (b) Centrally controlled

In the conventional scheme the nodes are distributed from root node. In the Centrally controlled scheme, the root nodes are distributed root nodes to sub nodes and then distributed.

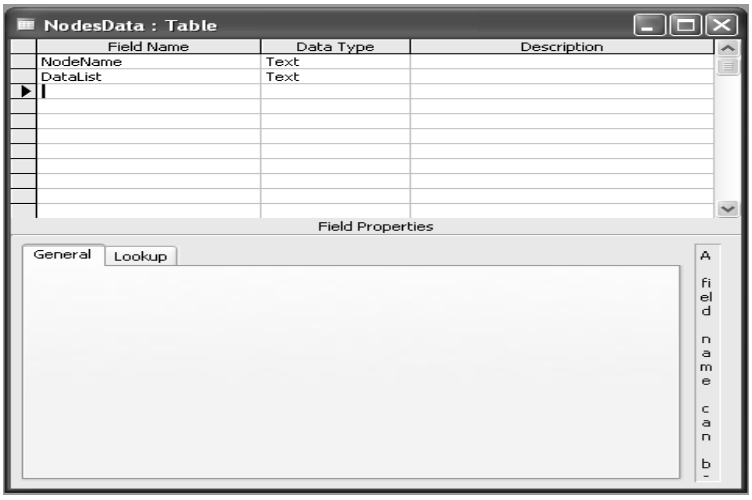


Table 1: storage of nodes as data

The nodes are stored in terms of data in order to locate the nodes which are decentralized.

5. Conclusion

We have considered the problem of K-means clustering on data horizontally distributed over a P2P network. Decentralizing all the data to a single location to run a centralized K-means is not a feasible option. Thus, the proposed work is without centralizing the data. Our algorithm works by having nodes communicate only with their topologically immediate neighbours in the network. This algorithm can adapt gracefully to a dynamic environment. Our experiments show it to produce highly accurate clustering results (relative to a centralized clustering). However, we cannot provide analytical guarantees on this clustering accuracy. we provide probabilistic guarantees on the clustering accuracy. And our experiments show the actual accuracy is quite good. In conclusion, we feel that our proposed algorithms are effective in solving a complicated data mining problem in large, distributed environment like a Peer-To-Peer wireless mess network. They provide good scalability and accurate results.

References

[1] P. Luo, H. Xiong, K. Lu, and Z. Shi, "Distributed Classification in Peer-to-Peer Networks," Proc. ACM Workshop Knowledge Discovery from Sensor Data (KDD '07), pp. 968-976, 2011.
[2] W. Kowalczyk, M. Jelasity, and A. Eiben, "Towards Data Mining in Large and Fully Distributed Peer-to-Peer Overlay Networks,"
[3] G. Forman and B. Zhang, "Distributed Data Clustering Can Be Efficient and Exact," SIGKDD Explorations, vol. 2, no. 2, pp. 34-38, 2000.
[4] W.T. Balke, W. Nejdl, W. Siberski, and U. Thaden, "Progressive Top-k Retrieval in Peer-to-Peer Networks," Proc. Int'l Conf. Data Eng. (ICDE '05), pp. 174-185, 2005.
[5] A. Vlachou, C. Doulkeridis, K. Norvag, and M. Vazirgiannis, "On Efficient Top-K Query Processing in Highly Distributed Environ-ments," Proc. ACM SIGMOD, pp. 753-764, 2008.
[6] S. Shi, J. Yu, G. Yang, and D. Wang, "Distributed Page Ranking in Structured P2P Networks," Proc. Int'l Conf. Parallel Processing (ICPP '03), pp. 179-186, 2003.
[7] S.Datta,K.Bhaduri,C.Giannella , R.Wolff and H.Kargupta, "Distributed Data Mining in Peer-to-Peer Networks", AGNIK LLC, Columbia, MD USA.
[8] M. Bawa, A. Gionis, H. Garcia-Molina, and R. Motwani, "The Price of Validity in Dynamic Networks," J. Computer and System Sciences, vol. 73, no. 3, pp. 245-264, 2007.

