Performance of Distributed Hierarchical Cluster in Peer to Peer Network Traffic

M.Vijayakumar1,†, R.M.S.Parvathi2

1 Research Scholar & Assistant Professor, Department of CSE, Sasurie College of Engineering, Tamilnadu, India
2 Principal & Professor, Department of CSE, Sengunthar College of Engineering for Women, Tamilnadu, India

Abstract
Distributed data mining in peer network traffic is emerging as a new distributed computing paradigm for many novel network applications that involve exchange of information among a large number of peers with less centralized coordination. Huge data sets are being collected regularly from the peer network traffic, but it is still extremely difficult to draw conclusions or make decisions based on the collective characteristics of such dynamically changing traffic volumes. The previous work based on distributed data clustering in peer network traffic involves a hierarchy of P2P neighborhoods. The peers in each neighborhood are responsible for building a clustering solution, using P2P communication, based on the accessible data. With the growing hierarchy, clusters are merged from lower levels in the hierarchy. At the root of the hierarchy, one global clustering can be derived. However, traffic nature of peer nodes in the communication are assumed to be normal i.e., static. In reality peer node movements increases the partition instability. In this paper distributed hierarchical clustering approach is introduced to handle the dynamic nature of peer network traffic and also it allows nodes to join and leave the network, by maintaining a balanced network in terms of partitioning and height. This leads a way to find the optimal network height for certain peer network applications with improved scalability. It allows merging and splitting of complete hierarchies and the clustering algorithm more global by allowing centroids to cross neighborhoods through higher levels. Clusters at lower level neighborhoods acts as function of higher level centroids. The distributed search for cluster centroids is guided by a cluster quality measure that estimates intra-cluster cohesiveness and inter-cluster separation. Experimentation is conducted to estimate the cohesiveness of cluster object to measure the quality of distributed hierarchical cluster to existing k-means cluster.

Keywords: Peer Networks; Distributed Data Mining; Hierarchical Cluster

1. Introduction
Analyzing massive data sets, which often span different sites, using traditional centralized approaches can be intractable. Distributed Data Mining (DDM) is being fueled by recent advances in grid infrastructures and distributed computing platforms. Huge data sets are being collected daily in different fields; e.g., retail chains, banking, biomedicine, astronomy, and so forth, but it is still extremely difficult to draw conclusions or make decisions based on the collective characteristics of such disparate data. Four main approaches for performing DDM can be identified. A common approach is to bring the data to a central site, then apply centralized data mining on the collected data. Such approach clearly suffers from a huge communication and computation cost to pool and mine the global data. In addition, we cannot preserve data privacy in such scenarios.

† Corresponding author.

Email addresses: tovijayakumar@gmail.com (M.Vijayakumar)

1553-9105/ Copyright © 2011 Binary Information Press
June, 2011
A smarter approach is to perform local mining at each site to produce a local model. All local models can then be transmitted to a central site that combines them into a global model [1], [2], [3]. Ensemble methods also fall into this category [4]. While this approach may not scale well with the number of sites, it is a better solution than pooling the data. Another smart approach is for each site to carefully select a small set of representative data objects and transmit it to a central site, which combines the local representatives into one global representative data set. Data mining can then be carried on the global representative data set [5], [6]. All previous three approaches involve a central site to facilitate the DDM process. A more departing approach does not involve centralized operation, and thus belongs to the peer-to-peer (P2P) class of algorithms. P2P networks can be unstructured or structured. Unstructured networks are formed arbitrarily by establishing and dropping links over time, and they usually suffer from flooding of traffic to resolve certain requests. Structured networks, on the other hand, make an assumption about the network topology and implement a certain protocol that exploits such topology. In P2P DDM, sites communicate directly with each other to perform the data mining task [7], [8], [9], [10].

Communication in P2P DDM can be very costly if care is not taken to localize traffic, instead of relying on flooding of control or data messages. Regardless of any particular DDM approach, the distributed nature of the DDM concept itself usually entails a tradeoff between accuracy and scalability. If better accuracy is desired, the granularity level of information exchanged between distributed nodes should become finer and/or the connectedness of different nodes should be increased. On the other hand, if better scalability is desired, the granularity should be coarser and/or the connectedness should be reduced. Bandyopadhyay [11] derive for their P2P K-means algorithm an upper bound on the clustering error during computing the distributed solution, which measures the degree to which accuracy has been sacrificed at the expense of lowered communication cost. The relation is evident in many DDM methods as well, since the prevalent approaches employ approximate algorithms, as opposed to exact algorithms.

In this paper, we introduce an approach for distributed data clustering, based on a structured P2P network architecture. The goal is to achieve a flexible distributed data mining model that can be tailored to various scenarios. The proposed model is a hierarchically distributed clustering for peer to peer networks traffic characteristics. It involves a hierarchy of P2P neighborhoods, in which the peers in each neighborhood are responsible for building a clustering solution, using P2P communication, based on the data they have access to.

2. Distributed Hierarchical Clustering in Peer Network Traffic

Distributed Hierarchical cluster in P2P architecture is used for analyzing network traffic characteristic scalability. Scalable distributed clustering system (or any data mining system for that matter) should involve hierarchical distribution[13][14][15]. A hierarchical processing strategy allows for delegation of responsibility and modularity. Central to this hierarchical architecture design is the formation of neighborhoods. A neighborhood is a group of peers forming a logical unit of isolation in an otherwise unrestricted open P2P network.

Peers in a neighborhood can communicate directly but not with peers in other neighborhoods. Each neighborhood has a super node. Communication between neighborhoods is achieved through their respective super nodes. This model reduces flooding problems usually encountered in large P2P
networks[16][17]. The notion of a neighborhood accompanied by a super node can be applied recursively to construct a multilevel overlay hierarchy of peers; i.e., a group of super nodes can form a higher level neighborhood, which can communicate with other neighborhoods on the same level of the hierarchy through their respective (higher level) super nodes.

The hierarchical distributed cluster is an iterative process in the peer network for the network traffic characteristic changing over the time[18]. It is a centroid-based clustering algorithm, where a set of cluster centroids is generated to describe the clustering solution. Each neighborhood converges to a set of centroids that describe the data set in that neighborhood. The distributed clustering strategy within a single neighborhood is similar to the parallel K-means algorithm in that the final set of centroids of a neighborhood will be identical to those produced by centralized K-means on the data within that neighborhood. Other neighborhoods, either on the same level or at higher levels of the hierarchy, may converge to another set of centroids. Once a neighborhood converges to a set of centroids, those centroids are acquired by the super node of that neighborhood. The super node, in turn as part of its higher level neighborhood, collaborates with its peers to form a set of centroids for its neighborhood. This process continues hierarchically until a set of centroids is generated at the root of the hierarchy. The DHP2PC is evaluated with entropy and separation index measures[12][13].

Fig. 1 Hierarchical Peer to Peer Clustering Architecture

2.1. Cluster Entropy Measure

Entropy reflects the homogeneity of a set of objects, and thus can be used to indicate the homogeneity of a cluster. Lower cluster entropy indicates more homogeneous clusters. On the other hand, measure the entropy of a pre-labeled class of objects, which indicates the homogeneity of a class with respect to the generated clusters. The less fragmented a class across clusters, the higher its entropy, and vice versa. This is referred to as class entropy. For every cluster \( c_j \) in the clustering result \( c \), compute the probability that a member of cluster \( c_j \) belongs to class \( l_i \). The entropy of each cluster \( c_j \) is calculated using the standard formula
\[ E_{c_j} = -\sum \frac{n(l_i,c_j)}{n(c_j)} \log \frac{n(l_i,c_j)}{n(c_j)}, \]

where the sum is taken over all classes. The total entropy for a set of clusters is calculated as the sum of entropies for each cluster weighted by the size of each cluster:

\[ E_c = \sum_{j=1}^{n(c)} \frac{n(c_j)}{n(D)} \times E_{c_j} \]

Cluster entropy rewards small clusters, which means that if a class is fragmented across many clusters it would still get a low entropy value. To counter this problem, calculate class entropy. The entropy of each class \( l_i \) is calculated using

\[ E_{l_i} = -\sum \frac{n(l_i,c_j)}{n(l_i)} \log \frac{n(l_i,c_j)}{n(l_i)}, \]

where the sum is taken over all clusters. The total entropy for a set of classes is calculated as the weighted average of the individual class entropies:

\[ E_l = \sum_{i=1}^{n(l)} \frac{n(l_i)}{n(D)} \times E_{l_i} \]

As with cluster entropy, a drawback of class entropy is that if multiple small classes are lumped into one cluster, their class entropy would still be small. To evaluate the overall entropy cluster and class entropy are combined into an overall entropy measure:

\[ E_c(\alpha) = \alpha \cdot E_c + (1 - \alpha) \cdot E_l \]

Evaluate the quality of clustering at different levels of the hierarchy. At level \( h=0 \), we evaluated the quality of clustering for each neighborhood, with respect to the subset of the data in the neighborhood, i.e.

\[ E_r = E_{c_{r \cap D^r}}, \]

where \( cr \) is the set of clusters obtained for neighborhood \( r \), and \( Dr \) is the union of data sets of all nodes in that neighborhood

\[ (D^r = U_{i \in q} D^i) \]

At level \( h > 0 \), evaluate the clustering acquired by a super node with respect to the data subset of the nodes at the level 0 reachable from the super node. Thus, evaluation of the clustering acquired at the root node reflects the quality with respect to the whole data set.

### 2.2. Separation Index

SI is another cluster validity measure that utilizes cluster centroids to measure the distance between clusters, as well as between points in a cluster to their respective cluster centroid. It is defined as the ratio of average within-cluster variance (cluster scatter) to the square of the minimum pair wise distance between clusters.
where \( m_i \) is the centroid of cluster \( c_i \), and \( \text{dist}_{\text{min}} \) is the minimum pairwise distance between cluster centroids. Clustering solutions with more compact clusters and larger separation have lower Separation Index, thus lower values indicate better solutions. This index is more computationally efficient than other validity indices, which is also used to validate clusters that are compact and well separated. In addition, it is less sensitive to noisy data.

3. Experimental Results on Distributed Hierarchical Clusters

A simulation environment was used for evaluating the DHP2PC algorithm compared to that of the K-Means P2P cluster model. During simulation, data were partitioned randomly over all nodes of the network. The number of clusters was specified to the algorithm such that it corresponds to the actual number of classes in each data set. A random set of centroids was chosen by each supernode, and the centroids were distributed to all nodes in its neighborhood at the beginning of the process. Clustering was invoked at level 0 neighborhoods and was propagated to the root of the hierarchy. Then we evaluate the effect of network size on clustering accuracy, the effect of scaling the hierarchy height, the quality of clustering at different levels within a single hierarchy, and the accuracy of distributed cluster summarization.

3.1. Cluster Quality

The distributed search for cluster centroids is guided by a cluster quality measure that estimates intracluster cohesiveness and intercluster separation. Cluster cohesiveness is the distribution of pairwise similarities within a cluster is represented using a cluster similarity histogram, which is a concise statistical representation of the cluster tightness. To estimate the cohesiveness of cluster calculate the histogram skew. Skew is the third central moment of a distribution, it tells us if one tail of the distribution is longer than the other. The positive skew indicates a longer tail in the positive direction (higher interval of the histogram), while a negative skew indicates a longer tail in the negative (lower interval) direction. The similarity histogram that is negatively skewed indicates a tight cluster.

3.2. Network Size and Height

Experiments on different network sizes and heights were performed, and their effect on clustering accuracy (Entropy and SI) and speedup over centralized clustering were measured. The first observation here is that for networks of height, the distributed clustering accuracy stays almost the same as the network size.
increases. This is evident through both the Entropy and SI. Since for networks of height 1 all nodes at level 0 are in the same neighborhood, every node can update its centroids based on complete information received from all other nodes at the end of each iteration (at the cost of increased communication). This means that increasing the network size does not affect accuracy of clustering, as long as it is of height 1.

The second observation is that, for networks of the same size, larger network heights cause clustering accuracy to drop. It is not surprising that this is the case, since at higher levels meta-clustering of lower level centroid is expected to produce some deviation from the true centroids. It is also noticeable that unlike networks of height 1, networks with height $H > 0$ tend to have less accuracy as the number of nodes is increased. This in turn means neighborhoods become smaller, thus causing the more accurate centroids at level 0 to become more fragmented.

An interesting observation is that there is a noticeable plateau region between the centralized case and a point where the data are finely partitioned, after which quality degrades rapidly. This plateau provides a clue on the relation between the data set size and the number of nodes, beyond which the number of nodes should not be increased without increasing the data set size. An appropriate strategy for automatically detecting the higher boundary of this region (in scenarios where the network grows arbitrarily) is to compare the SI measure before and after adding nodes; if a sufficiently large difference in SI is noticed then network growth should be suspended until more data are available (and equally partitioned). Since the increase in hierarchy height has the biggest effect on the accuracy of the resulting clustering accuracy, a strategy based on the SI measure can be adopted to select the most appropriate hierarchy for a certain application.

The experiment investigates the effect of increasing hierarchy heights, as well as the accuracy at different levels within a single hierarchy, in more detail in the next sections. In terms of speedup, the trends show that the HP2PC algorithm exhibits decent speedup over the centralized case. And also speedup scale well with the network size, by optimizing increased communication cost for networks of that height. This result carries an assertion that the hierarchical architecture of HP2PC is indeed scalable compared to flat P2P networks.

### 3.3. Performance Comparison of Distributed Hierarchical Cluster with K-Means in Peer Networks

The accuracy of distributed hierarchical cluster is compared with P2P K-means, which is the current state of the art in P2P-based distributed clustering. Since the implementation of P2P K-means is nontrivial, benchmark synthetic data set are used to compare against.

The above data set is a mixture of Gaussians distribution, containing 2,000 points. The actual data were not available from the authors but rather the parameters of the Gaussians, which we used to regenerate the data.

The measure of accuracy was based on the difference between cluster membership produced by P2P K-means and that of the same data point as produced by the centralized K-means. To ensure accurate comparison, initial seeds for both the centralized and the P2P algorithms were the same. The report shows that the total number of mislabeled data points as a percentage of the size of the data set. Fig 2 shows the results for P2P K-means and distributed hierarchical cluster (with various hierarchy heights). Nodes vary between 50 and 500.
<table>
<thead>
<tr>
<th>n(p)</th>
<th>P2P K-Means</th>
<th>D-HP2PC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H=1</td>
<td>H=2</td>
</tr>
<tr>
<td>50</td>
<td>1.87</td>
<td>0.00</td>
</tr>
<tr>
<td>100</td>
<td>1.77</td>
<td>0.00</td>
</tr>
<tr>
<td>150</td>
<td>1.93</td>
<td>0.00</td>
</tr>
<tr>
<td>200</td>
<td>1.84</td>
<td>0.00</td>
</tr>
<tr>
<td>250</td>
<td>1.67</td>
<td>0.00</td>
</tr>
<tr>
<td>300</td>
<td>2.15</td>
<td>0.00</td>
</tr>
<tr>
<td>350</td>
<td>2.68</td>
<td>0.00</td>
</tr>
<tr>
<td>400</td>
<td>3.98</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Fig. 2 Comparison between D-HP2PC and P2P K-Means

Fig. 3 Comparison of Distributed Hierarchical Cluster Vs P2P K-Means

Fig 3 illustrates the trend in the results. Distributed hierarchical cluster has zero error for networks of height 1. It is clear that for networks of low height, it is superior to P2P K-means. As the height increases, proposed model starts to approach the error rate of P2P K-means, but interestingly, it does not suffer from the sharp increase in percentage mislabeled peers at very large number of nodes. P2P K-means has an advantage of being a model for unstructured P2P networks. It assumes that each node has a finite number of reachable neighbors, which are randomly selected from the node population. HP2PC, on the other hand, has a fixed hierarchical structure that allows it to produce superior results by avoiding random peering and propagation delay and error, a common disadvantage in P2P networks.

4. Conclusion

The proposed model presented distributed hierarchical cluster for peer network traffic characteristic evaluation, which allows building hierarchical networks for clustering data. The experimentation demonstrated the flexibility of the model, showing that it achieves comparable quality to its centralized
counterpart while providing significant speedup and that it is possible to make it equivalent to traditional distributed clustering models (e.g., facilitator-worker models) by manipulating the neighborhood size and height parameters. The model shows good scalability with respect to network size and hierarchy height, degrading the distributed clustering quality significantly.

The importance of this contribution stems from its flexibility to accommodate regular types of P2P networks with dynamic peer nodes as well as modularized networks through neighborhood and hierarchy formation. It also allows privacy within neighborhood boundaries (no data shared between neighborhoods). In addition, we provide interpretation capability for document clustering through document cluster summarization using distributed key phrase extraction. The proposed one this model to be dynamic, allowing nodes to join and leave the network, by maintaining a balanced network in terms of partitioning and height. This will also lead us to a way to find the optimal network height for certain applications. The clustering algorithm is made more global by allowing centroids to cross neighborhoods through higher levels centroids. This is achieved with better global clustering solutions but on the expense of computational complexity. The future improvement made in the direction of extending it to allow merging and splitting of complete hierarchies.

References


