Improving the Performance of Banking Sector by Using Clustering Method: An Object – Oriented Approach

Dr. Harsh Dev
Pranveer Singh Institute of Technology, 544-555
NH -2, Kalpi Road, BhaiutiKanpur, UP, India
Email: doctorharshdev@rediffmail.com

Suman Kumar Mishra
Department of Computer Science, Bhagwant University,
Ajmer, Rajasthan, India
Email: sumansunil_532@yahoo.com

ABSTRACT
In present scenario, high performance cluster-based web server is needed to be deployed by banking services to fit the ever – increasing demands of the online banking users. In the last few years, the increased pressure of high-online banking users have overloaded the existing web clusters and thus they fail to supply better services to all the online users resulting in unexpected long delays. To handle this situation, Object Oriented approach needs to be implementing in the web clusters. This research proposes how to improve the performance of the data clustering by using the object - oriented approach.

Keywords: Cluster-based web server, Banking services, Better services, Object Oriented approach

1. INTRODUCTION
Clustering gets its name as a metaphor for the data mining. Cluster systems have been widely accepted as a cost effective solution for various applications such as banking systems. However, online banking users experience long delays when retrieving pages from the cluster systems. Actually, in partitions a set of unlabeled data, objects are divided into different groups and within groups, there exist different cluster, each cluster containing similar data objects. Clustering aims at dividing data sets into subsets called clusters. Web clusters have been accepted by banking systems as a cost effective solution for various applications such as online banking services and database management. However, online banking users experience unpredictable delays when retrieving web pages from the cluster systems. The object - oriented approach with web clusters can manage the Delay Problems.

Web cluster analysis has been widely used in different applications such as data analysis, market research and pattern recognition. There are many ways to improve performance in Database Management System. Clustering, Buffering and Indexing are the main techniques used in conventional Database Management System.

In an OODB, “classes are used as a primary means of grouping objects and searching for objects. In some sense, classes take the place of relations. There is an assumption that there must be system-supported mechanisms for grouping objects into collections automatically”[1].

Advanced database applications like Computer-Aided Design (CAD), Computer-Aided Manufacturing (CAM), Computer-Aided Software Engineering (CASE), Office Information System (OIS), Multimedia System, Geographic Information Systems (GIS), and Interactive and Dynamic Web Sites are focusing heavily on the use of OODBs technology as the process integration framework. This is due to the OODBMSs that can manage complex and highly interrelated information. The Relational Databases (RDBs) have showed some shortcomings in managing this type of information [2].

The different clustering approaches can be classified with the help of the hierarchy [3] given below in Figure 1.

![Figure 1: Taxonomy of Clustering](image-url)
2. HIERARCHICAL CLUSTERING

Hierarchical clustering algorithm is mainly used for datasets to form decomposes based structure and does not require determining the number of clusters in advance. It can be broadly divided into two types, agglomerative hierarchical clustering and divisive hierarchical clustering. In agglomerative hierarchical clustering, used bottom up approach each data is treated as a separate cluster. An agglomerative approach starts with n clusters that is by considering each point in the dataset as a cluster and in iterative loops apply merge operation to reach to root node, which is a cluster containing all data objects. The merge operation has based on the distance between two clusters. There are two different notions of distance: single link and complete link.

In divisive hierarchical clustering data, points split into number of cluster based on certain criteria-using top down approach. A divisive method, opposite to agglomerative of starts with a root node and considering all data objects into a single cluster, and in successive steps tries to divide the dataset until reaches to a leaf node containing a single object. For a dataset having n objects there is \( 2^{n-1} - 1 \) possible two-subset divisions, which is very expensive in computation. Two divisive clustering algorithms, DIANA and MONA [4][5].

Static clustering approaches depend on having the whole collection ready before applying the algorithm. Dynamical algorithms have the ability to update the clustering when data are added or removed from the collection [6].

3. PARTITIONAL CLUSTERING

Partitional clustering techniques attempt to directly partition data objects into a set of disjoint clusters [7]. Partitional clustering is a combinatorial problem, which is a branch of discrete optimization problems. In addition, in partitional clustering, the set of feasible solutions is finite and grows combinatorially with the problem size [8]. The partitional clustering aims at optimizing cluster centers and the number of clusters. One of the major drawbacks of the partitional approaches is the difficulty in determining the number of clusters [9]. In many clustering problems, the correct number of clusters is not known, and it is impossible to estimate. Most clustering algorithms need to determine the number of clusters in advance. A solution for this problem is to use dynamic clustering techniques. Dynamic clustering techniques have two general objectives, finding the optimal number of clusters and partitioning the data objects into clusters [10].

4. OBJECT – ORIENTED DATABASE

At present scenario, use of object – oriented approach in various fields of information technology has necessitated object oriented database system. Object – Oriented Databases (OODBs) have been designed to support large and complex data queries. The data accuracy, consistency and integrity in OODBs are extremely important for programmers and designer. The basic approach of OODBMSs is to extend and implement the objectives of object – oriented programming to the database management domain. The extension of object – oriented programming concepts to a database management environment necessitates the provision of additional characteristics, necessary to support storage and manipulation of large, shared and persistent objects. These characteristics, typical to the database environment, include support mechanisms and indexing schemes for large secondary storage organizations of a database, concurrency control and recovery [11]. In given table comparison between Relational database and Object Oriented database Management System [12]:

<table>
<thead>
<tr>
<th>No.</th>
<th>Functions</th>
<th>RDBMS</th>
<th>OODBMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Main objectives</td>
<td>Data Encapsulation and independence.</td>
<td>Ensuring data independence from application programs.</td>
</tr>
<tr>
<td>2.</td>
<td>Independen- ced of classes</td>
<td>Classes can be reorgan-ized without affecting the mode of using them.</td>
<td>Support New data types (Class)</td>
</tr>
<tr>
<td>3.</td>
<td>Complex Objects</td>
<td>Hierarchical data structure</td>
<td>Support Retrieval as a single unit</td>
</tr>
<tr>
<td>4.</td>
<td>Version control</td>
<td>No support old version</td>
<td>Support previous and next version attributes</td>
</tr>
<tr>
<td>5.</td>
<td>Schema evaluation</td>
<td>No support efficient mechanism</td>
<td>Changes to the definition inside a class and changes to the structure of the class lattice.</td>
</tr>
<tr>
<td>6.</td>
<td>Equivalent objects</td>
<td>Do not provide mechanisms to Model semantics of equivalent representations.</td>
<td>Different representation s of the same object</td>
</tr>
<tr>
<td>7.</td>
<td>Long transaction s</td>
<td>Returns to the most recent state possible - not just to the state at the last committed transaction.</td>
<td>Support for versions can be integrated with transaction management to simplify concurrent access and recovery.</td>
</tr>
</tbody>
</table>
5. DATA CLUSTERING TECHNIQUES

5.1 K- Means Clustering

The K- means clustering, is an algorithm based on finding data clusters in a data set such that a cost function of dissimilarity measure is minimized [12]. A set of m vectors $y_i$, $i = 1, 2, 3, \ldots, m$, are to be partitioned into d groups $G_j$, $j = 1, 2, \ldots, d$. The cost function, based on the Euclidean distance between a vector $y_m$ in group $i$ and the corresponding cluster center $d_j$, can be defined by:

$$I = \sum_{j=1}^{d} I_j = \sum_{j=1}^{d} \left( \sum_{m, y_m \in G_j} \|y_m - d_j\| \right)$$

5.2 K- Means++ Clustering

Let $D(x)$ denote the shortest distance from a data point to the closest center we have already chosen. Then, define the following algorithm [14]:

(i) Take one center $c_1$, chosen uniformly at random from $\chi$.

(ii) Take a new center $c_i$, choosing $x \in \chi$ with probability

$$\frac{D(x)^2}{\sum_{x \in \chi} D(x)^2}$$

(iii) Repeat step ii until we have taken k centers altogether.

5.3 Subtractive Clustering

In this method, the computation is proportional to the problem size instead of the problem dimension.

$$S_j = \sum_{i=1}^{m} \exp \left( - \frac{\|y_j - y_i\|^2}{(\tau_d/2)^2} \right)$$

6. IMPLEMENTATION AND RESULTS

Having discussed the different clustering based techniques and object – oriented database features, we now move to the discussion of these techniques and features based on a practical approach. These approach involves the implementation of three clustering techniques (K – means, K- means ++ and subtractive), and testing each one of them on a set of banking data related to Fixed deposit with different maturity years and interest rates. The banking data consist of six input attributes related to Rate of interest on Fixed deposits and one output attribute which indicates whether the customers specimen signature matches or not. The data set consist of 100 cases. The data set is partitioned into two clusters; i. e. customer specimen signature matches at withdrawal time, and customer specimen signature not matches at withdrawal time.
We can test the effect of the two different variables $v_1$ and $v_2$ on the accuracy of the subtractive algorithm. In the above figure, the result shows poor accuracy due to either very small or very large changes in output.

7. CONCLUSION

In this paper, three clustering techniques (K-means, Subtractive and K-means++) have been reviewed, implemented and tested against a data set of banking system for their fixed deposits using object-oriented approach. The comparative analysis of accuracy and performance measures for three clustering techniques has been done on Versant software. Banking system involves high number of dimensions with complicated relationship between the variables in the input data. But in this case, using K-means clustering to look over the performance of the other clustering techniques but when number of clusters is not known, and then K-means cannot handle this type of problem. Subtractive clustering is used to solve this type of problem. When number of Grid Point increases in subtractive clustering, then high computation is needed for the larger number of grid points and it increases the complexity of data clustering. After implementation of K-means++ clustering with object-oriented database to decrease complexity for result and increase accuracy also. The same work can also be extended by combing of two algorithms and to take advantages of the features of both the algorithms. This approach can be used for the insurance, telecom and retail sector databases.

REFERENCES


means++: The Advantages of Central Seeding”,

[15] Bahmani Bahman, Moseley Benjamin, Vattani Andrea, Kumar Ravi, Vessilvitskii Sergei,

“Scalable K – means++” VLDB Endowment,