Replenish Approach in Non Homogeneous Structured Dataset using Interpolation Techniques

V. Narayani, S.Raj Kumar

Abstract— Replenish approach refers to the behavior of filling the fill the gaps in a table. Suppose that one has a table listing the population of some country in 1970, 1980, 1990 and 2000, and that one wanted to estimate the population in 1994.It lead us to implement the Numerical methods scxenario to solve this issue.The basic operation of linear interpolation between two values is so commonly used in computer graphics that it is sometimes called alerp in that field's jargon. The term can be used as a verb or noun for the operation. e.g. "Bresenham's algorithm lerps incrementally between the two endpoints of the line.". The behaviors in Distributed Database environment are joining a relation, sharing resources, extraction on queries, etc. we aim to learn to predict the missed datum in distributed database. The connections in this environment are not homogenous. To address the interdependency among data instances, relational learning has been proposed, and collective inference based on network connectivity is adopted for prediction. However, the connections in distributed database are often multi-dimensional. The heterogeneity presented in network connectivity can hinder the success of collective inference. Interpolation-based approach has been shown effective in addressing the heterogeneity of connections presented in distributed database system. The scale of these networks entails scalable learning of models for replenish prediction. This scheme is very sensitive to handle heterogeneity of distributed database system.In this paper we aim to predict the heterogeneity of two different environments by applying the interpolation schema which result the expected tuples. This method improves the performance of data extraction. Handling the heterogeneity of all distributed environments will be the future work.

Index Terms — Data mining, DDBMS, Interpolation, Replenish, Replication.

I. INTRODUCTION

A distributed database is a database in which storage devices are not all attached to a common processing unit such as the CPU,controlled by a distributed database management system (together sometimes called a distributed database system) . It may be stored in multiple computers, located in the same physical location; or may be dispersed over a network of interconnected computers [1]. Unlike parallel systems, in which the processors are tightly coupled and constitute a single database system, a distributed database system consists of loosely-coupled sites that share no physical components [2]. System administrators can distribute collections of data (e.g. in a database) across multiple physical locations.

A distributed database can reside on network servers on the Internet [3], on corporate intranets or extranets, or on other company networks. Because they store data across multiple computers, distributed databases can improve performance at end-user worksites by allowing transactions to be processed on many machines, instead of being limited to one [4].Two processes ensure that the distributed databases remain up-to-date and current: replication and duplication.

1. Replication involves using specialized software that looks for changes in the distributive database [5]. Once the changes have been identified, the replication process makes all the databases look the same. The replication process can be complex and time-consuming depending on the size and number of the distributed databases [6]. This process can also require a lot of time and computer resources.
2. Duplication, on the other hand, has less complexity. It basically identifies one database as a master and then duplicates that database [7]. The duplication process is normally done at a set time after hours. This is to ensure that each distributed location has the same data [8]. In the duplication process, users may change only the master database. This ensures that local data will not be overwritten.

Both replication and duplication can keep the data current in all distributive locations [9].

Besides distributed database replication and fragmentation, there are many other distributed database design technologies. For example, local autonomy, synchronous and asynchronous distributed database technologies [10]. These technologies' implementation can and does depend on the needs of the business and the sensitivity/confidentiality of the data stored in the database, and hence the price the business is willing to spend on ensuring data security, consistency and integrity [11]. When discussing access to distributed databases, Microsoft favors the term distributed query, which it defines in protocol-specific manner as "[a]ny SELECT, INSERT, UPDATE, or DELETE statement that references tables and rowsets from one or more external OLE DB data sources". Oracle provides a more language-centric view in which distributed queries and distributed transactions form part of distributed SQL.

II. SYSTEM DESIGN FOR DDBMS

1) Homogeneous DDBMS

In a homogeneous distributed database all sites have identical software and are aware of each other and agree to cooperate in processing user requests. Each site surrenders part of its autonomy in terms of right to change schema or software. A homogeneous DDBMS appears to the user as a single system.

Manuscript received March, 2014.

Dr.V.Narayani, Associate Professor, Dept. of MCA, Karpagam College of Engineering, Coimbatore, Tamilnadu, India.

Dr.S.Rajkumar,Assistant Professor (SG), Dept. of CSE , SNS College of Engineering, Coimbatore, Tamilnadu, India.

The homogeneous system is much easier to design and manage. The following conditions must be satisfied for homogeneous database:

- The operating system used, at each location must be same or compatible.
- The data structures used at each location must be same or compatible.
- The database application (or DBMS) used at each location must be same or compatible.

2) Heterogeneous DDBMS

In a heterogeneous distributed database, different sites may use different schema and software. Difference in schema is a major problem for query processing and transaction processing. Sites may not be aware of each other and may provide only limited facilities for cooperation in transaction processing. In heterogeneous systems, different nodes may have different hardware & software and data structures at various nodes or locations are also incompatible. Different computers and operating systems, database applications or data models may be used at each of the locations. For example, one location may have the latest relational database management technology, while another location may store data using conventional files or old version of database management system. Similarly, one location may have the Windows NT operating system, while another may have UNIX. Heterogeneous systems are usually used when individual sites use their own hardware and software. On heterogeneous system, translations are required to allow communication between different sites (or DBMS). In this system, the users must be able to make requests in a database language at their local sites. Usually the SQL database language is used for this purpose. If the hardware is different, then the translation is straightforward, in which computer codes and word-length is changed. The heterogeneous system is often not technically or economically feasible. In this system, a user at one location may be able to read but not update the data at another location.

III. METHODOLOGY

In mathematics, **linear interpolation** is a method of curve fitting using linear polynomials.

Linear Interpolation between two known points

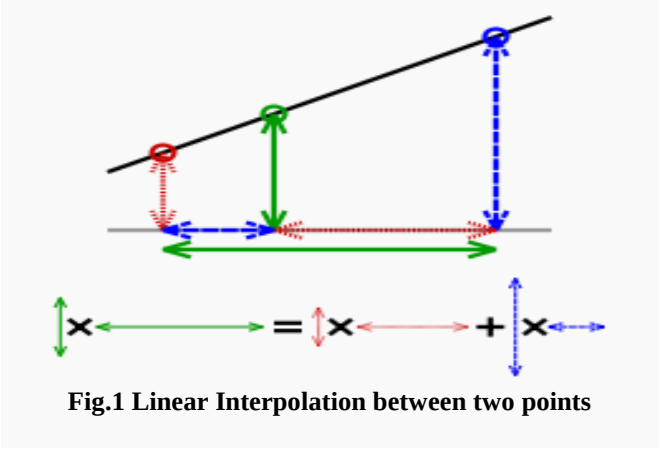

Fig.1 Linear Interpolation between two points

In this geometric visualisation, the value at the green circle multiplied by the distance between the red and blue circles is equal to the sum of the value at the red circle multiplied by the distance between the green and blue circles, and the value at the blue circle multiplied by the distance between the green and red circles.

If the two known points are given by the coordinates (x_0,y_0) and (x_1,y_1), the **linear interpolant** is the straight line between these points. For a value *x* in the interval (x_0,x_1), the value *y* along the straight line is given from the equation

$$\frac{y - y_0}{x - x_0} = \frac{y_1 - y_0}{x_1 - x_0} \quad \text{----------(1)}$$

which can be derived geometrically from the figure on the right. It is a special case ofpolynomial interpolation with *n* = 1.

Solving this equation for *y*, which is the unknown value at *x*, gives

$$y = y_0 + (y_1 - y_0)\frac{x - x_0}{x_1 - x_0} \quad \text{-----(2)}$$

which is the formula for linear interpolation in the interval (x_0,x_1). Outside this interval, the formula is identical to linear extrapolation.

This formula can also be understood as a weighted average. The weights are inversely related to the distance from the end points to the unknown point; the closer point has more influence than the farther point. Thus, the weights

$\frac{x - x_0}{x_1 - x_0}$ and $\frac{x_1 - x}{x_1 - x_0}$, which are normalized distances between the unknown point and each of the end points.

Interpolation as a Data set

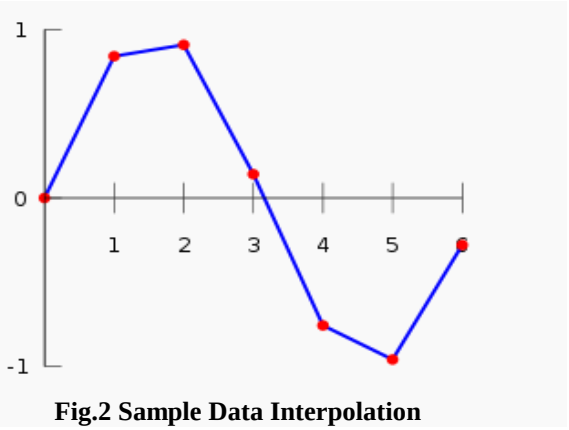

Fig.2 Sample Data Interpolation

Linear interpolation on a data set (red points) consists of pieces of linear interpolants (blue lines).

Linear interpolation on a set of data points (x_0, y_0), (x_1, y_1), ..., (x_n, y_n) is defined as the concatenation of linear interpolants between each pair of data points. This results in a continuous curve, with a discontinuous derivative (in general), thus of differentiability class C^0.

Linear interpolation as approximation

Linear interpolation is often used to approximate a value of some function *f*using two known values of that function at other points. The *error* of this approximation is defined as

$$R_T = f(x) - p(x) \quad \text{----------(3)}$$

where *p* denotes the linear interpolation polynomial defined above

$$p(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0). \quad \text{------(4)}$$

It can be proven using Rolle's theorem that if f has a continuous second derivative, the error is bounded by

$$|R_T| \leq \frac{(x_1 - x_0)^2}{8} \max_{x_0 \leq x \leq x_1} |f''(x)|. \quad \text{----(5)}$$

As you see, the approximation between two points on a given function gets worse with the second derivative of the function that is approximated. This is intuitively correct as well: the "curvier" the function is, the worse the approximations made with simple linear interpolation.

IV. PROPOSED ARCHITECTURE

The architecture diagram of our proposed system focusses with the implementation of interpolation techniques homogeneous or heterogeneous data which takes the prediction analysis in linear evaluation whereas the structured or unstructured datum takes the prediction analysis in non linear fashion.

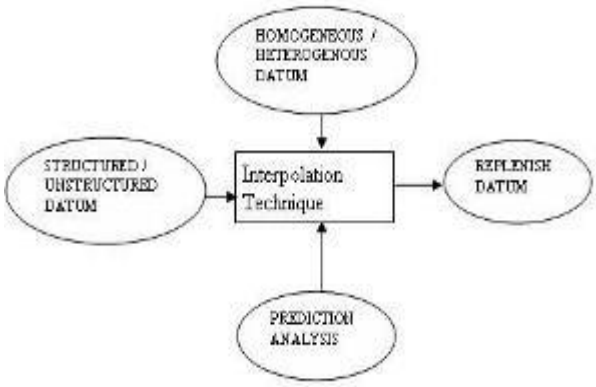

Fig.3 Proposed Architecture

The prediction analysis for local file system and remote filesystem are using the prediction analysis in a linear way whereas the images and videos utilizes the prediction analysis with data extraction,cleaning,feature analysis,pattern classification,clustering and categorization includes nonliner fashion.

V. IMPLEMENTATION

In a ABC Engineering college which maintains the Database Management system for the NBA accreditation purpose ,the number of computer systems required and check for the server count towards more than three years is also essential.If the data is inadequate then we must implement our proposed schema of interpolatrion and find the missed data and search for the appropriate bills,receipts and documents for the complete entirety towards efficient documentation.Consider the sample data as follows,such that the missing year of 2013 is a necessary data for the NBA process.

Table 1:Interpolation computation table

Sl.No	Accreditation Year	No of Systems in LAN for IT department	No of servers Maintained in IT Department
1	Jan-2011	102	07
2	Jan-2012	215	11
3	Jan-2013	312	?
4	Jan-2014	409	21

Applying the formula (1)

Therefore
X0=215,y0=11,x1=409,y1=21,x=312
y-11/x-215=10/194,y=0.052*312=16.08
Hence the total no of servers available during that period was 16 servers.

VI. RESULTS AND DISCUSSION

The final results we achieved through the proposed technique is as follows,

Table 2: Interpolation Prediction Table

Sl.No	AccreditationYear	No of Systems in LAN for IT department	No of servers Maintained in IT Department
1	Jan-2011	102	07
2	Jan-2012	215	11
3	Jan-2013	312	16
4	Jan-2014	409	21

The number of servers in the particular period was correct and we identified the corresponding bill from the huge collection of ambiguous cluster of receipts.The final results are shown below,

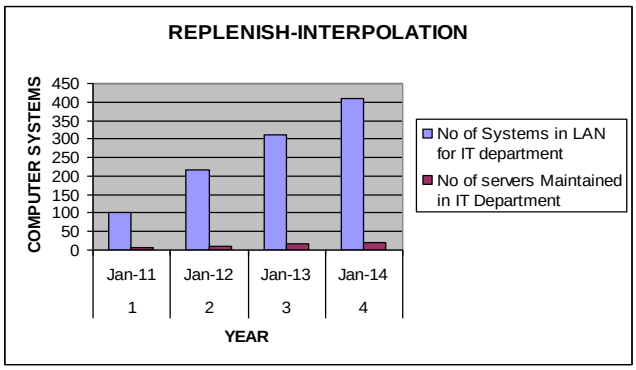

Fig.4 Proposed Model Implementation

VII. CONCLUSION

The heterogeneity presented in network connectivity can hinder the success of collective inference. Interpolation-based approach has been shown effective in addressing the heterogeneity of connections presented in distributed database system. The scale of these networks entails scalable learning of models for replenish prediction. This scheme is very sensitive to handle heterogeneity of distributed database system.In this paper we identified the missing datum using Interpolation techniques which peformed the Replenish approach refers to the behavior of filling datum in non homogeneous structured datum.In near future we will implememt the Genetic algorithm apporoach for the replenish techniques in unstructured datum also.Our proposed strategy produced good results which will be initiated further tuning towards efficient data prediction system.

REFERENCES

1. G. W. Leibniz, C. I. Gerhardt, "Historia et origo calculi differentialis", Mathematische Schriften, vol. 5, pp.392 -410 1971 :Georg Olms Verlag

2. J. M. Child, W. J. Greenstreet, "Newton and the art of discovery", Isaac Newton 1642&ndash,1727: A Memorial Volume, pp.117 -129 1927 :G. Bell
3. J. Simpson and E. Weiner, The Oxford English Dictionary, 1989 :Oxford Univ. Press
4. J. Bauschinger, W. F. Meyer, "Interpolation", Encyklopä,die der Mathematischen Wissenschaften, pp.799 -820 1900&ndash,1904 :B. G.Teubner
5. J. Wallis, Arithmetica Infinitorum, 1972 :Olms Verlag
6. T. N. Thiele, Interpolationsrechnung, 1909 :B. G. Teubner
7. O Neugebauer, A History of Ancient Mathematical Astronomy, Springer-Verlag
8. G. J. Toomer, C. C. Gillispie and F. L. Holmes, "Hipparchus", Dictionary of Scientific Biography, vol. XV, pp.207 -224 1978 :Scribner
9. G. van Brummelen, "Lunar and planetary interpolation tables in Ptolemy',s Almagest", J. Hist. Astron., vol. 25, no. 4, pp.297 -311 1994
10. D. Maluf, D. Bell, N. Ashish, C. Knight and P.Tran, "Semi-structured data management in the enterprise: A nimble, high-throughput, and scalable approach," in International Database Engineering and Applications Symposium (IDEAS), 2005
11. Maluf, David A., Tran, Peter B., "NETMARK: A Schemal-Less Extension for Relational Databases for Managing Semi-Structured Data Dynamically," International Symposium on Methodologies for Intelligent Systems, Lecture Notes in Computer Science, Springer Verlag, 2003.

AUTHORS PROFILE

Dr.S.RAJKUMAR completed his M.E Computer Science and Engineering in 2004 and Ph.D in 2013 respectively. His area of interest is soft computing. He published more than 25 international journals in the area of soft computing.