



Fog Formation Prediction In Coastal Regions Using Data Mining Techniques

Nelson F. F. Ebecken

COPPE/UFRJ - Federal University of Rio de Janeiro

Caixa Postal 68506

21945-970 Rio de Janeiro - Brazil

e-mail: nelson@ntt.ufrj.br

Abstract

The recent growth of many scientific and business databases created a need of a new class of tools and techniques for automated search the knowledge hidden in these databases. For this reason it became necessary to integrate data mining and database technology on scalable parallel platforms.

This paper discusses a parallel implementation of a data mining strategy developed for MIMD platforms: objected related data mining technology, that enables a specific form of data mining optimized for better performance and more complete results.

Our application is related to observed meteorological conditions at the International Airport of Rio de Janeiro. The main objectives are to understand the parameters that have influence on visibility/weather conditions (fog formations), and to compare with current methodologies to predict future situations with some antecedence. In our analysis we have considered the last 20 years to build a model and extract the knowledge to make predictions.

1 Introduction

With recent explosion of information, and the availability of cheap storage, it has been possible to gather massive data during the last decades. The eventual goal of this massive data gathering is the use of this information to gain competitive advantages, by discovering previously unknown patterns in data, which can guide the decision making.

Recent years have shown that it is becoming difficult to analyze data using only OLAP tools, showing the need of an automated process to discover interesting and hidden patterns in data. Data mining techniques have increasingly been studied e.g. Mark¹, especially in their application in real-world databases. One typical problem is that databases tend to be very large, and these techniques often repeatedly scan the entire set. A solution that has been used for a long time is the extraction of a training sample, so that the data mining algorithm could be run on a smaller portion of data. In this case, subtle differences among sets of objects in the database will become less evident.

The need to handle large amounts of data implies a lot of computational power, memory and disk I/O, which can only be provided by parallel computers.

A variety of data mining algorithms have been proposed addressing parallel processing. Most of them use flat files and exploit parallelism through parallel programming.

2 The Data Mining Strategy

In the present work the neural approach was used. A large number of papers have been presented: e.g., Lu & Liu², Bigus³, Schurmann⁴. The neural network based data mining approach consists of three major phases:

- *Network construction and training.* This phase constructs and trains a three layer neural network based on the number of attributes and number of classes and chosen input coding method.
- *Network pruning.* The pruning phase aims at removing redundant links and units without increasing the classification error rate of the network. A small number of

units and links left in the network after pruning enable us to extract concise and comprehensible rules.

- *Rule extraction.* This phase extracts the classification rules from the pruned network.

The rule extraction algorithm used was suggested by Lu & Liu⁵ and can be summarized as:

- Apply a clustering algorithm to find clusters of hidden node activation values.
- Enumerate the discretized activation values and compute the network outputs. Generate rules that describe the network outputs in terms of the discretized hidden unit activation values.
- For each hidden unit, enumerate the input values that lead to them and generate a set of rules to describe the hidden units' discretized values in terms of the inputs.
- Merge the two sets of rules obtained in the previous two steps to obtain rules that relate the inputs and outputs.

3 Scalable Parallel System

Our parallel algorithms have been implemented on an IBM POWER parallel System SP from NACAD, the High Performance Computing Nucleus of COPPE/UFRJ, the Engineering Graduate Center of the Federal University of Rio de Janeiro, with the following characteristics:

- 2 wide nodes (spn1 and spn2): RAM memory 1GB(spn1)/512MB(spn2), Disk 8GB, processor type POWER2 model 590.
- 2 thin nodes (spn3 and spn4): RAM memory 128MB, Disk 2GB, processor type POWER2 model 390.
- TOTAL: 1064Mflops, RAM 1.75Gbytes, Disk 20Gbytes.



168 Environmental Coastal Regions

The parallel implementation of the backpropagation algorithm has two approaches:

- partitioning the training data sets, that distributes the data sets over nodes, and
- the partitioning of the network itself, that distributes the neurons over processing nodes.

The approach for partitioning the data has the advantage of reducing the required communication to produce the network output. Each node executes the forward and backward passes independently on a different subset of total data. This leads to a performance gain and low complexity of the implementation. The network may be copied in each processing node and perform the training over its training data set.

In a MIMD architecture, each node can have different power processing, local memory and disks. The mapping algorithm of processes to nodes must take into account these differences and choose the suitable procedure of distribution. Another issue of this implementation is the workload balance on processing nodes. On systems where dynamic workload balance is present the mapping algorithm should use this feature to specify the proceedings. But it's hard to determinate the best parameter to be used in remapping the network, leading us to a complexity growth and some undesired situations.

Our first implementation works with static mapping of nodes and partitioning of the training data sets, keeping copies of the entire network on each node.

4 Oracle Parallel Server

The present work was conducted under ORACLE Parallel Server environment.

Oracle software was installed in CWS (Control Workstation). A nfs mounted directory (/oracle) is then imported by each of the nodes of RISC/6000 SP2.

Environmental Coastal Regions 169

The advantages are: centralized installation (easier to do software upgrades) and it saves disk storage needed, but it does not offer high availability, because the CWS represents a single point of failure.

There is a single database, which is accessible from any of the SP2 nodes (transparent to users). The files which store the database are distributed among all the 4 nodes of the system. All datafiles are accessible from all nodes through the VSD (Virtual Shared Disk), a device driver layer above the AIX Local Volume Manager. To guarantee data consistency, each node of the system runs an Oracle software called DLM (Distributed Lock Manager). The DLM is responsible for locking services of shared resources of the system (in this case, the datafiles).

To improve performance, Oracle Parallel Server uses a Parallel Cache (each node has its own local data cache, which is available from all other nodes through DLM and HPS).

The Parallel Query Option provides scalability to most common query operations. It dramatically improves response times for very large complex queries. Scans, sorts, joins, distinct and groups can all be performed in parallel.

The application developed to access data in parallel is exactly the same that would access data sequentially (transparent parallelism to application). Oracle Parallel Server requires all the disks in the system to be accessible from all nodes (equivalent to a Shared Disk approach). When a query is processed in parallel, Oracle knows which data are on the local disks of a given node. The nodes which are local to the requested data are primarily used for query processing to increase performance.

The parallel processing can be decomposed in three phases:

Phase I: Shared-Everything - the query is divided into subtasks, which are sent to the nodes where data reside.

Phase II: Shared-Disk - each of the processing nodes does purely local processing to retrieve the requested data.

Phase III: Shared-Disk - nodes which have finished early their processing can be borrowed to take over tasks from nodes which have not finished processing yet.



170 Environmental Coastal Regions

The workload can be redistributed dynamically since all nodes have access to all data. The processing power of all nodes can be used to process the query as effectively as possible.

Oracle Parallel Server for the RS/6000 SP provides a number of advanced system administration tools, making system management of an MPP system as easy as any other system. System administration tasks can be performed from a single point of control.

5 The Studied Application

Our application is related to observed meteorological conditions at the International Airport of Rio de Janeiro. The main objectives are to understand the parameters that have influence on visibility/weather conditions (fog formations), and to compare with current methodologies to predict future situations with some antecedence.

In our analysis we have considered the last 20 years to build a model and extract the knowledge to make predictions. The weather condition is stored in the 12th field, and it is varying in the range 0-9. The fog condition corresponds to the value 4.

In this way our problem can be stated as:

- What are the relevant parameters and rules that imply in value 4?
- What are the relevant parameters and rules, occurred “n” hours before, that imply in value 4?

The database stores for each hour (24 hours per day) the following values:

- | | |
|----------------------|---|
| 1. register code | 19. clouds at 3 rd level: number |
| 2. register block | 20. clouds at 3 rd level: type |
| 3. register location | 21. clouds at 3 rd level: height |
| 4. year | 22. clouds at 4 th level: number |
| 5. month | 23. clouds at 4 th level: type |
| 6. day | 24. clouds at 4 th level: height |

- | | |
|---|---------------------------------------|
| 7. hour | 25. altimeter correction |
| 8. clouds total | 26. temperature (Dew-Celsius) |
| 9. wind direction | 27. blast velocity |
| 10. wind velocity | 28. atmospheric pressure - sea level |
| 11. horizontal visibility | 29. atmospheric pressure-runway level |
| 12. weather condition | 30. cloud direction |
| 13. clouds at 1 st level: number | 31. pressure trend |
| 14. clouds at 1 st level: type | 32. pressure - last 3 hours variation |
| 15. clouds at 1 st level: height | 33. air temperature - dry |
| 16. clouds at 2 nd level: number | 34. air temperature humid |
| 17. clouds at 2 nd level: type | 35. precipitation |
| 18. clouds at 2 nd level: height | 36. relative humidity |

6 Results

The primary metric for evaluating classifier performance is classification accuracy: the percentage of test samples that are correctly classified. The other important metrics is classification time (speed-up) and the structure of the extracted knowledge (accurate and compact). The obtained error is 5.96%, and the sensitivity analysis is shown in Table 1.

A great number of informations were extracted by rules. The fog prediction can be characterized by:

$$IF\{(-7796 \leq 21.x1 + 32.x2 + x3 - 14.x4 - 84.x5 \leq -5880)$$

and

$$-10460 \leq 27.x1 + x3 - 25.x4 - 107.x5 \leq -8060)\}$$

These results will be confronted to current practice methodology and evaluated by expert opinion.

The obtained speed-up corresponds to a performance factor of 2.2 between the 4 nodes solution and the best 1 node solution.



172 Environmental Coastal Regions

Table 1. Sensitivity

Field Name	Sensitivity
Precipitation	42.9
Hour	9.0
Clouds 2 nd level type	5.1
Wind direction (x2)	4.8
Temperature (dew) (x4)	4.4
Clouds 2 nd level: number	4.3
Pressure-sea level	3.4
Pressure trend	3.4
Pressure 3hours variation	3.3
Month (x1)	3.1
Clouds 1 st level height	3.0
Wind velocity	2.9
Clouds total	2.9
Clouds 1 st level number	2.6
Clouds 1 st level type	2.0
Clouds 2 nd level height (x3)	0.6
Relative humidity (x5)	0.5

7 Conclusion

This work is an attempt to study the connectionist approach to data mining to generate classification similar to that of decision trees. The main issue is to reduce the training time of neural networks, developing fast algorithms and database implementations on parallel scalable architectures.

Although the parallel environment permits a fast initial training of a network, it is desirable to have incremental training and rule extraction during the life time of an application database: the accuracy of rules extracted can be improved along the change of database contents. Another possibility to improve the performance is to reduce the number of input units of the networks by feature selection.

The present data mining technology adds the following improvements:

- Better performance and reduced cycle times because compute-intensive tasks are executed in parallel, and closer to the data source.
- The analysis is more interactive and involves fewer process steps when performed online, directly against the relational data source.
- Ability to mine much larger data sets, not just the data in flat files or extracts..
- More complete results - without the need to extract a target data set, all potentially valuable information is available for mining.

References

- [1] Mark, W., Special Issue on Data Mining, IEEE Expert, Vol. 11, n^o. 5, October, 1996.
- [2] Lu, H., Setiono, R. and Liu, H., Neurorule: A Connectionist Approach to Data Mining, Proc. VLDB'95, pp 478-489, 1995.



174 Environmental Coastal Regions

- [3] Bigus, J. P., Data Mining with Neural Networks, McGraw-Hill, 1996.
- [4] Schurmann, J., Pattern Classification: A Unified View of Statistical and Neural Approaches, John Wiley, 1996.
- [5] Lu, H., Setiono, R. and Liu, H., Effective Data Mining Using Neural Networks, IEEE Transactions on Knowledge and Data Engineering, Vol. 8, no. 6, December, 1996.
- [6] Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R., editors, Advances in Knowledge Discovery and Data Mining, AAAI Press/MIT Press, 1996.
- [7] Agrawal, R. and Shafer, J. C., Parallel Mining of Association Rules, IEEE Transactions on Knowledge and Data Engineering, Vol. 8, no. 6 December, 1996.
- [8] ORACLE Parallel Server, ORACLE Cooperation, 1996.