Grid Data Mining for Outcome prediction in Intensive Care Medicine

Manuel Filipe Santos, Wesley Mathew, and Carlos Filipe Portela

Centro Algoritmi, Dep. Sistemas de Informação, Universidade do Minho, Portugal
{mfs,wesley,cfp}@dsi.uminho.pt

Abstract. This paper introduces a distributed data mining approach suited to grid computing environments based on a supervised learning classifier system. SCM and MVM methods for Distributed Data Mining (DDM) are explored and compared with the Centralized Data Mining (CDM) approach. Experimental tests were conducted with a real world data set from the intensive care medicine in order to predict the outcome of the patients. The results demonstrate that the performance of the DDM methods are better than the CDM method.

Keywords: Intensive Care Medicine, Outcome Prediction, Distributed Data Mining, Centralized Data Mining.

1 Introduction

Recently, there is a significant progress in the research related to distribute data mining. Digital data stored in the distributed environments is doubling within a few years. More advanced and feasible distributed data mining algorithms and strategies are required in the current fast growing environment.

Learning Classifier System (LCS) is a concept formally introduced by John Holland as a genetic based machine learning algorithm [1]. Manuel Santos [2] developed the DICE system, a parallel and distributed architecture for LCS. In his work he attempted to parallelize the genetic algorithm and LCS message operations to increase system’s performance. A. Giani, Dorigo and Bersini also did significant re attained in the experimental work research in the area of parallel LCS [3]. Their implementation also tried to increase the performance of the system. All implementations of parallel LCS consider a single data and generate a single model.

This work is part of two major projects – the Gridclass project – whose main goal is to implement the UCS in a grid environment and – the INTCare project – whose main goal is to implement an intelligent decision support system for Intensive Care Units where the data distribution among distinct sites is an important issue. Gridclass system does not paralyze any part of the UCS. Various instances of the UCS are executed in different distributed sites with different set of data. All the experimental work was done using the Grid gain platform; a java based distributed computing middleware [4].

The key objective of this work is to construct a global data mining model from different local models of the grid and compare DDM and CDM methods. Grid computing architecture is considered the best distributed framework for solving the distributed data mining task [5, 6]. Each node of the grid environment executes different UCS and those nodes send local data mining models to the central site for developing a global model. This paper introduces two different methods for merging local models from each distributed sites. The different strategies are: Specific Classifier Method (SCM), Majority Voting Method (MVM).

The Intensive Medicine is a specific environment where the patients normally are in weak conditions. The decisions are normally mad by some stress or by a necessity of quickly response. For the doctors is very difficult make decision in this conditions especially when they don't have the required clinical data about the patients. In order to help them some projects were created and INTCare [7, 8] is one of them. One of the main goals of INTCare is the outcome prediction in Intensive Care Units. In order to meet this objective, a new platform was developed that allows the clinical data collect in real-time and in electronic format. This data will used in a distributed data mining approach suited to grid computing environments based on a supervised learning classifier system.

Remaining sections of this paper are organized as follows: Section 2 gives the background details of the intensive care unit data, section 3 describes the way of data acquisition from ICU and section 4 explains the global model construction methods such as specific classifier method and majority voting method. Section 5 shows the experimental set up and results of DDM and CDM. Section 6 discusses the performance of DDM vs. CDM. Further section 6 shows some related works and final section presents main conclusions.

2 Background

2.1 Intensive Care Units

The Intensive Care Units (ICU) is the place where the knowledge and treatments associated Intensive Medicine is applied. The main purposes of ICU are diagnose, monitor and treat patients with serious illnesses and recover them for their health and quality of life prior [9]. ICUs are concerned with these patients, and focus their efforts on the resuscitation of patients who are terminally ill or in treating patients who are vulnerable to an organic dysfunction, benefiting from the preventive care for each system dysfunction according to the principles of restoration to normal physiology [10], maintaining a serious and continuous monitoring of the patient. In the ICUs are used as decision support systems the disease severity index and prediction models, to predict the risk of in-hospital mortality through a set of prognostic variables that uses the predictive index of disease severity [11]. The models predict the mortality risk for a number of patients with a certain degree of physiological dysfunction. The most famous outcome prediction index is SAPS that is based on the worst results recorded in the first 24 hours after admission [12]. The Systems that use this type of indices usually selects the patient, evaluates and records the predictor variables, calculates the severity index and returns the rate of mortality.

3 Data Acquisition in ICU

3.1 KDD Process in ICU

In order to obtain the maximum number of electronic data we develop an Electronic Nursing Record (ENR) that integrate a high number of hospital data sources like Electronic Health Process (EHR), lab results, allow data acquisition, data monitoring and data validation, electronically, online and in real-time. After the data be collected, these will be prepared and transformed to be used in distributed data mining approach suited to grid computing. The Fig. 1 shows the data sources and the Knowledge Discovery in Database (KDD) process used in the ICU.

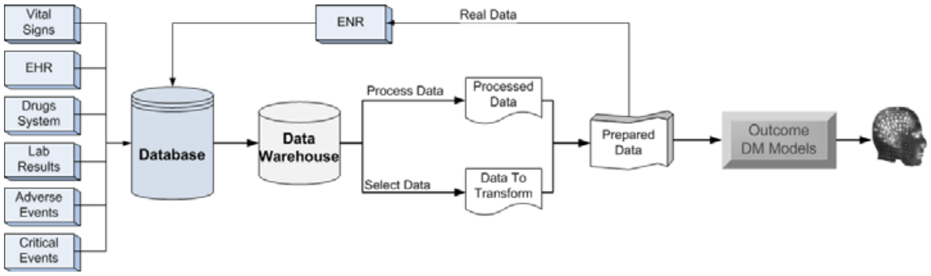

Fig. 1. ICU Knowledge Discovery in Database Process

3.2 Data Set Description

The data used in this approach were collected in real-time and were related with patient who had an entire stay with a full monitoring in ICU in the firsts five days. This data correspond to three months and thirty two patients. The input variables consist of: Admission data; Critical Events (CE); SOFA; and Accumulated Critical Events (ACE). The admission data (i.e. age, admission type and admission from) and Critical Events (CE), derived from four physiologic variables Blood pressure (BP), heart rate (HR) and oxygen saturation (SPo2) that were collected by the bedside monitors and urine output (UR) [13] . The Table 1 presents the values that are in the dataset and are obtained at the patient admission and after patient discharge.

Table 1. Possible values of patient admission data

Variable	Description	Range
Hour	relating to 5 days of stay	[1-120]
Age	The age of patient admitted in ICU	1 - [18; 46]; 2- [47; 65]; 3 - [66; 75]; 4 - >= 76
Admission Type	The type of admission	{Urgent (U); Programmed (P)};
Admission From	Admission origin of the patient	1 - Surgery block, 2 - Recovery room, 3 - Emergency room, 4 - Nursing room, 5 - Other ICU, 6 -

		Other hospital, 7 - Other sources
Outcome	Patient final discharge	{Survivor (0); Deceased (1)}

For each variable (VAR): BP, HR, SPo2 and HR were calculated the AEC, EC and a set of ratios, the Table 2 show the descriptions of each ratio and the possible values. CE was defined by a panel of experts [14]. If a physiological parameter is out of its normal range [15] for more than 10 minutes or the result is lower than the minimum acceptable, it is considered a CE. In consequence of CE we have the Accumulated Critical Events (ACE) that was derived as a new variable and is an hourly sum of CE of one patient during its staying.The score used in this data set was SOFA, which can quantify the level of failure (0-4) to each organ system (neurologic, cardiovascular, hepatic, renal, respiratory, coagulation). In this case, we transformed the data and considered 0 to normal values and 1 if an organ failure happened.

Table 2. Possible values of events, ratios, and scores

Variable	Description	Range
EC	Number of critical events of each VAR occurred per hour	[0; + ∞]
AEC	Number of accumulated critical events of each VAR occurred	[0; + ∞]
ec_ac_var / **EC_max**	Number of accumulated critical events of each VAR occurred Maximum number of critical events possible in an hour	[0; 1]
ec_ac_var / **Horas**	Number of accumulated critical events of VAR occurred Hours of stay	[0; 1]
tot_ec_ac	Number of total critical events accumulated of all 4 variables	[0; + ∞]
tot_ec_ac / **ec_max**	Number of total critical events accumulated of all 4 variables Maximum number of critical events possible in an hour of all var	[0; 1]
tot_ec_ac / **Horas**	Number of total critical events accumulated of all 4 variables Hours of stay	[0; 1]
sofa_organ	SOFA value for each organ system	Failure (1) Normal (0)

Incorrect values were detected and corrected by ignoring values considered absurd by the medical experts. The resulting data of this prepared data process were used by Data Mining.

4 Global Model Construction

Gridclass uses the UCS for data mining proposes. Two levels of data mining models are generated in the Gridclass system. The first level is related to the models generated in each distributed sites and the second level correspond to the model generated in the central site. The first data mining models are known as local models. The second level is known as global model and is generated from all the local models in the first level. The global model represents all the data in the distributed environment.

During the training process, Gridclass system generates data mining models based on the training data and a predefined set of classifier [16]. If a predefined set of classifiers is provided, then the system can perform incremental learning. The incremental learning process improves the performance therefore the system can provide more generalized learning model. If a predefined set of classifiers is not provided, then the system generates the data mining models only from training data. Data mining models are maintained by genetic algorithm and covering operations in UCS system [17, 18, 19].

There are many challenges for constructing a global model, because wrong combination of the classifiers gathered from the local models, will affect negatively the performance of global model. The main difficulty is to derive the significance of each classifier and predict their values in the global model. All training data are completely independent even though there should be many similar classifiers with different sets of parameter values (benefits). Therefore the parameter evaluation of the classifiers in the global model is important.

Remaining sections demonstrate some solutions that are suitable for constructing the global model. Each strategy establishes different sort of combinations of local models in the global model. Those strategies help to understand the significance of availability of different sort of local classifiers in the global model. Each strategy has peculiar significance for the development of the global model. The performance of global model is evaluated from the testing accuracies of the global model

4.1 Specific Classifier Method (SCM)

Specific Classifier Method (SCM) only preserves discrete classifiers in the global model [20]. SCM induce the global model without repeating similar classifiers and simultaneously keeping all the benefits of the local classifiers.

In SCM the initial process is to collect all the classifiers from the distributed sites and store them in a central location. The collected classifiers have to be evaluated based on the criteria of SCM and those classifiers that are eligible to be integrated the global model will be stored in the global model. While classifiers are evaluated, each classifier needs to be matched with all other classifiers in the collected local model. When one classifier finds another similar classifier in the collected local models then that classifier updates its parameters with parameters of matched classifier. Finally, the induced global model will be tested using a data set that was generated from the global data set.

4.2 Majority Voting Method (MVM)

Majority Voting Method (MVM) is another strategy for constructing the global model from distributed local models. The goal of the MVM is to eradicate weak classifiers from the global model and construct a strong model in the central system (global model). Initially, MVM gathers all local models and stores them in the central system, then goes on to find all discrete classifiers from the accumulated local models as SCM. Later, the system calculates a threshold value (*cut_ off_ threshold*) from the collected classifiers and uses it to benchmark the classifiers in the population [20]. If the accuracy of a classifier is greater than the *cut_ off_ threshold* value then that classifier will be stored in the global model.

5 Experimental work

Experimental work intents to compare the performance of DDM and CDM therefore different sizes of iteration, population size and node are considered in the distributed site. ICU data set has 3570 records of data and each record has 31 fields and each field has different ranges of the values. ICU data was divided for training and testing, i.e. randomly selected 70% of original data was considered as centralized training data and randomly selected 30% of original data was considered as centralized testing data. For the DDM training and testing data was made from the centralized training and centralized testing datasets. Based on the number of nodes in the distributed site centralized training and centralized testing data was equally divided. Centralized training dataset has 2380 records and centralized testing dataset has 1190. Two set of nodes were considered (Ten and twenty) in the distributed site therefore for 10 nodes 238 records of data in each training dataset and 119 records of data in each testing dataset. For the 20 nodes tests, 119 records of data were considered in each training dataset and 59 records of in each testing dataset. Similarly, considerable size of population and number of iterations of the CDM, population size and number of iterations were divided according to the number of nodes in the DDM. Three sets of iterations were considered for CDM that are 100000, 200000 and 300000 and four set of population sizes were selected for CDM that are 500, 1000, 2000 and 4000. For the ten nodes in the DDM considered iterations are 10000, 20000 and 30000 and considered populations are 50, 100, 200 and 400. For the twenty nodes, considered iterations are 5000, 10000 and 15000 and considered population sizes were 25, 50, 100 and 200. To compare the performance of each approach, we considered the accuracies (the average of 10 executions). The configuration parameters used in the UCS are: *ProbabilityOfClassZero* = 0.5, *V* = 20, *GaThreshold* = 25, *MutationProb* = 0.05, *CrossoverProb* = 0.8, *InexperienceThreshold* = 20, *InexperiencePenalty* = 0.01, *CoveringProbability* = 0.33, *ThetaSub* = 20, *ThetaSubAccuracyMinimum* = 0.99, *ThetaDel* =20, *ThetaDelFra* = 0.10.

5.1 DDM Experiments

Table 3 shows the global model testing accuracies attained for the SCM and MVM strategies. Based on the testing accuracies, it is difficult to say which the best method for constructing the global model. But based on the global population size MVM is the best because the global population size of the MVM is always smaller than the global population size of the SCM. Testing accuracies increase in proportion to the population size as expected, for example, almost 71% of accuracy is achieved with global population size of 500, near to 80% of accuracy is achieved with global population size of 1000, approximately 87% of accuracy is achieved with global population size of 2000, and nearly 93% of accuracy is achieved with global population size of 4000. Higher population sizes were not considered in order to avoid overfitting phenomena.

Table 3. Testing accuracies of global models generated using SCM and MVM.

Number of Nodes	Iterations	Local Population Size	Accuracy		Global Population Size	
			SCM	MVM	SCM	MVM
10	10,000	50	0.716 ± 0.0110	0.7132 ± 0.01252	485.8 ± 4.87	381.3 ± 10.187
10	10,000	100	0.7987 ± 0.01586	0.7987 ± 0.0175	955 ± 5.35	655.7 ± 9.2141
10	10,000	200	0.8784 ± 0.01715	0.876 ± 0.01511	1884.8 ± 12.23	1070.8 ± 20.48
10	10,000	400	0.925 ± 0.009	0.92606 ± 0.0088	3730.9 ± 17.615	1710.7 ± 33.40
10	20,000	50	0.7116 ± 0.0203	0.723 ± 0.0318	486.4 ± 3.687	383.2 ± 9.635
10	20,000	100	0.80 ± 0.0159	0.807 ± 0.0217	958.8 ± 7.08	648.2 ± 11.698
10	20,000	200	0.8794 ± 0.060	0.8722 ± 0.01589	1885 ± 11.72	1067.5 ± 21.36
10	20,000	400	0.925 ± 0.0099	0.9229 ± 0.0123	3724 ± 12.18	1713 ± 42.62
10	30.000	50	0.712 ± 0.018	0.7188 ± 0.0151	484 ± 2.366	382.5 ± 12.020
10	30,000	100	0.807 ± 0.0173	0.8024 ± 0.0167	958.7 ± 4.80	654.8 ± 10.695
10	30.000	200	0.875 ± 0.019	0.8723 ± 0.0179	1890.2 ± 9.96	1063. ± 31.287
10	30,000	400	0.9244 ± 0.0085	0.925 ± 0.01153	3720.1 ± 20.82	1705.5 ± 24.24
20	5,000	25	0.7203 ± 0.0192	0.7345 ± 0.0232	488.2 ± 3.119	394.1 ± 6.789
20	5,000	50	0.8028 ± 0.0176	0.797 ± 0.0177	959.1 ± 6.55	676.1 ± 17.47
20	5,000	100	0.879 ± 0.0186	0.8781 ± 0.01084	1890 ± 11.2570	1111.9 ± 28.68

20	5,000	200	0.932 ± 0.0130	0.927 ± 0.00674	3733 ± 14.2126	1779.7 ± 31.16
20	10,000	25	0.72 ± 0.018	0.721 ± 0.0158	486 ± 4.13	391.7 ± 6.412
20	10,000	50	0.805 ± 0.0192	0.8061 ± 0.0197	962.6 ± 4.501	669.3 ± 16.97
20	10,000	100	0.8824 ± 0.0167	0.884 ± 0.0151	1892 ± 9.04	1101.8 ± 16.87
20	10,000	200	0.9298 ± 0.0153	0.9369 ± 0.0118	3729.9 ± 13.194	1757.3 ± 33.50
20	15,000	25	0.7197 ± 0.0965	0.7158 ± 0.0212	486.6 ± 4.5509	389.6 ± 7.29
20	15,000	50	0.8091 ± 0.0129	0.8054 ± 0.0134	961.6 ± 7.381	673.2 ± 38.473
20	15,000	100	0.8695 ± 0.0135	0.8699 ± 0.0132	1886 ± 10.286	1110.7 ± 10.69
20	15,000	200	0.9325 ± 0.00977	0.9325 ± 0.01022	3738.8 ± 20.339	1777.6 ± 019.18

5.2 CDM Experiments

The testing accuracies of the CDM are smaller than the testing accuracies of DDM. The testing accuracies of the CDM also show the impact of the population size because the testing accuracies are increasing proportional to the population size.

Table 4. Testing accuracies for the CDM method.

Iteration	Population Size	Accuracy
100,000	500	0.56232 ± .17046
100,000	1000	0.6035 ± 0.182586
100,000	2000	0.6585 ± 0.1992
100,000	4000	0.7086 ± 0.2138
200,000	500	0.565 ± 0.170825
200,000	1000	0.5974 ± 0.1808
200,000	2000	0.64885 ± 0.1962
200,000	4000	0.7114 ± 0.2146
300,000	500	0.5585 ± 0.1689
300,000	1000	0.5996 ± 0.1814
300,000	2000	0.6507 ± 0.1965
300,000	4000	0.7156 ± 0.216

6 Discussion and Related work

The main goal of this work is to induce global data mining models and compare the performance of CDM versus the DDM methods. Two strategies described above were able to construct the global model from the distributed local models. The global model in the CDM method is obviously representing the overall problem (dataset) in the distributed site because that model is generated from global data without any intervention. Global model of the DDM and learning model of the CDM were tested with the same data. Though table 1 and 2 shows that DDM have best accuracies than CDM. For example, 500 global population size of the DDM has around 71% of testing accuracy but the testing accuracy of the 500 population size of the CDM is about 55%, 1000 global population size of the DDM has around 80% of testing accuracy but the testing accuracy of the 1000 population size of the CDM is about 60%, 2000 global population size of the DDM has around 87% of testing accuracy but the testing accuracy of the 500 population size of the CDM is about 65% and 4000 global population size of the DDM has around 93% of testing accuracy but the testing accuracy of the 500 population size of the CDM is about 71%.

7 Conclusions and Future Work

This paper presented the performance of CDM and DDM approaches using ICU real data in order to predict the outcome of critical care patients. The experimental results clearly show that the performance of the DDM is better than the performance of CDM. The DDM strategies of SCM and MVM achieved similar testing accuracies but the global population size of MVM is smaller than the global population size of the SCM. The results are very important in areas were distributed data should be considered without discharging the local models induction as is the ICU.

Further work will include more methods to construct the global models from the distributed local learning models.

ACKNOWLEDGMENTS: The authors would like to express their gratitude to FCT (Foundation of Science and Technology, Portugal), for the financial support through the contracts GRID/GRI/81736/2006 and INTCare - PTDC/EIA/72819/2006.

References

1. M. F. Santos, W. Mathew, T. Kovacs, H. Santos, A grid data mining architecture for learning classifier system. WSEAS TRANSACTIONS on COMPUTERS Volume 8, 2009 ISSN: 1109-2750
2. M. F. Santos. Learning Classifier System in Distributed environments, University of Minho School of Engineering Department of Information System. PhD Thesis work 1999.
3. Giani, Parallel Cooperative classifier system. Dottorato di ricerca in informatica Universita di Pisa, PhD Thesis TD-4/ 99.
4. http://www.gridgain.com/key_features.html. Consulted on 8 - 2 - 2011.

5. M.Cannataro, A. Congiusta, A. Pugliese, D.Talia, P. Trunfio, Distributed Data Mining on Grid: Services, Tools, and Applications. *IEEE TRANSACTIONS ON SYSTEM, MAN, AND CYBERNETICS- PART B: CYBETNETICS, VOL. 34 NO6,* DECEMBER 2004

6. J. Luo, M. Wang, J. Hu, Z. Shi, Distributed data mining on Agent Grid: Issues, Platform and development toolkit. *Future Generation computer system 23 (2007) 61-68*

7. Santos, M.F., Portela, F., Vilas-Boas, M., Machado, J., Abelha, A., Neves, J.: INTCARE - Multi-agent approach for real-time Intelligent Decision Support in Intensive Medicine. 3rd International Conference on Agents and Artificial Intelligence (*ICAART*). Springer, Rome, Italy (2011)

8. Gago, P., Santos, M.F., Silva, Á., Cortez, P., Neves, J., Gomes, L.: INTCare: a knowledge discovery based intelligent decision support system for intensive care medicine. Journal of Decision Systems (2006)

9. Suter, P., Armaganidis, A., Beaufils, F., Bonfill, X., Burchardi, H., Cook, D., Fagot-Largeault, A., Thijs, L., Vesconi, S., Williams, A.: Predicting outcome in ICU patients. Intensive Care Medicine 20, 390-397 (1994)

10. Hall, J.B., Schmidt, G.A., Wood, L.D.H.: Principles of Critical Care. McGraw-Hill's AccessMedicine (2005)

11. Silva, Á.: Modelos de Inteligência Artificial na análise da monitorização de eventos clínicos adversos, Disfunção/Falência de órgãos e prognóstico do doente critico. Ciências Médicas, vol. Doutoramento, pp. 281. Universidade do Porto, Porto (2007)

12. Le Gall, J.R., Lemeshow, S., Saulnier, F.: A new Simplified Acute Physiology Score (SAPS II) based on a European/North American multicenter study. JAMA 270, 2957-2963 (1993)

13. Vilas-Boas, M., Santos, M.F., Portela, F., Silva, Á., Rua, F.: Hourly prediction of organ failure and outcome in intensive care based on data mining techniques. In: Springer (ed.) 12th International Conference on Enterprise Information Systems, pp. 9, Funchal, Madeira, Portugal (2010)

14. Silva, Á., Pereira, J., Santos, M., Gomes, L., Neves, J.: Organ failure prediction based on clinical adverse events: a cluster model approach. In: Conference Organ failure prediction based on clinical adverse events: a cluster model approach. ACTA Press, (Year)

15. Silva, Á., Cortez, P., Santos, M.F., Gomes, L., Neves, J.: Rating organ failure via adverse events using data mining in the intensive care unit. Artificial Intelligence in Medicine 43, 179-193 (2008)

16. J. Luo, M. Wang, J. Hu, Z. Shi, Distributed data mining on Agent Grid: Issues, Platform and development toolkit. *Future Generation computer system 23 (2007) 61-68.*

17. http://www.idsia.ch/~juergen/icmlkolmogorov/node9.html. Consulted on 1/7/ 2010.

18. H. H. Dam, A scalable Evolutionary Learning Classifier System for Knowledge Discovery in Stream Data Mining, M.Sci. University of Western Australia, Australia, B.Sci. (Hons) Curtin University of Technology, Australia. *Thesis work 2008.*

19. Orriols-Puig, A Further Look at UCS Classifier System. *GECCO'06,* July 8–12, 2006, Seattle, Washington, USA

20. M. F. Santos, W. Mathew, and H. Santos: GridClass: Strategies for Global Vs Centralized Model Construction in Grid Data Mining, Proceeding of the workshop on ECAI, Lisbon 2010.