Grid Data Mining Strategies for Outcome Prediction in Distributed Intensive Care Units

Manuel Filipe Santos[1], Filipe Portela[1], Miguel Miranda[2], José Machado[2], António Abelha[2], Álvaro Silva[3], Fernando Rua[3]

[1]Centro Algoritmi, [2]CCTC
Universidade do Minho, Portugal,
[1]{mfs, cfp}@dsi.uminho.pt
[2]{miranda, jmac, abelha}@di.uminho.pt

[3] Serviço de Cuidados Intensivos, Centro Hospitalar do Porto
Largo Prof. Abel Salazar, Porto, Portugal
moreirasilva@clix.pt, fernandorua.sci@hgsa.min-saude.pt

ABSTRACT

Previous work developed to predict the outcome of patients in the context of intensive care units brought to the light some requirements like the need to deal with distributed data sources. Those data sources can be used to induce local prediction models and those models can in turn be used to induce global models more accurate and more general than the local models. This paper introduces a distributed data mining approach suited to grid computing environments based on a supervised learning classifier system. Five different tactics are explored for constructing the global model in a Distributed Data Mining (DDM) approach: Generalized Classifier Method (GCM); Specific Classifier Method (SCM); Weighed Classifier Method (WCM); Majority Voting Method (MVM); and Model Sampling Method (MSM). Experimental tests were conducted with a real world data set from the intensive care medicine. The results demonstrate that the performance of DDM methods is very competitive when compared with the centralized methods.

KEYWORDS: Intensive Care Medicine, Outcome Prediction, Distributed Data Mining, Grid Computing, Centralized Data Mining.

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 (M. F. Santos, Mathew, Kovacs, & Santos, 2009). Manuel Santos (Manuel Filipe Santos, 1999) 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 (Giani, Starita, & Vanneschi, 1999). 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 (Gain, 2006).

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 (Luo, Wang, Hu, & Shi, 2007; M.Cannataro, 2004). 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 work considers five different methods for merging local models from each distributed sites (M. F. Santos, et al., 2009; M. F. Santos, Mathew, & Santos, 2010 ; M. F. Santos, Mathew, & Santos, 2011). The different strategies are: Specific Classifier Method (SCM), Weighted Classifier Method (WCM), Generalized Classifier Method (GCM), Majority Voting Method (MVM) and Model Sampling Method (MSM).

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 (Gago et al., 2006; Manuel Filipe. Santos et al., 2011) 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 and INTCare, section 3 describes the way of data acquisition from ICU and section 4 explains the global model construction methods. 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.

BACKGROUND

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 (Suter et al., 1994). 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 (Hall, Schmidt, & Wood, 2005), maintaining a serious and continuous monitoring of the patient.

In the ICUs, decision support systems are mainly used for disease severity scoring and prediction modelling, to predict the risk of in-hospital mortality through a set of prognostic variables that uses the predictive index of disease severity (Álvaro Silva, 2007). 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 the Simplified Acute Physiology Score (SAPS) that is based on the worst results recorded in the first 24 hours after admission (Le Gall, Lemeshow, & Saulnier, 1993). The systems that use this type of indices usually select the patient, evaluate and stores the predictor variables, calculate the severity index and return the rate of mortality.

Intcare

INTCare is a research project whose main objective is to implement an Intelligent Decision Support System (IDSS) to predict the dysfunction or failure of six organic systems and the patient outcome in order to help doctors, in real-time, deciding on the better treatments or procedures for the patient (Gago, 2008). The ICU systems provide high volumes of data from different and complex data sources, like is, for example: bedside monitors, electronic health records, electronic nursing records, laboratory results and pharmacy drugs systems. INTCare makes use of ICU data to predict clinical situations.

All data is collected in real-time and pre-processed automatically by agents that are present in INTCare System (Manuel Filipe Santos, et al., 2011). The agents are autonomous and are associated to some tasks of the INTCare modules: Data Acquisition, Knowledge Management, Knowledge Inference and Interface. The flexibility and effectiveness of such systems depend on the agents and the interactions among them. In the context of this work have been used the agents: Vital Signs Acquisition, Gateway, ENR Agent, Pre-Processing and AIDA. INTCare system is pervasive in nature (Varshney, 2009), because the information, essential to the decision making, is available anywhere and anytime. The main features (Portela, Santos, Silva, Machado, & Abelha, 2011) of the system can be grouped in terms of:

- **Online Learning -** The system acts online, i.e., the DM models are induced using online data in opposition of an offline approach, where the data is gathered and processed afterwards;
- **Real-Time -** The system actuates in real-time, for the data acquisition and storing is made immediately after the events take place to allow that decisions are taken whenever an event occurs;
- **Adaptability -** The system has the ability to, automatically, optimize the models with new data when needed. This information is obtained from their evaluation results;
- **Data mining models -** The success of IDSS depends, among others, on the acuity of the DM models, i.e., the prediction models must be reliable. These models make it possible to predict events and avert some clinical complications to the patients;
- **Decision models -** The achievement of the best solutions depend heavily on the decision models created. Those are based in factors like differentiation and decision that are applied on prediction models and can help the doctors to choose the better solution on the decision making process;
- **Optimization** – The DM models are optimized over time. With this, their algorithms are in continuous training so that increasingly accurate and reliable solutions are returned, improving the models acuity;
- **Intelligent agents -** This type of agents makes the system work through autonomous actions that execute some essential tasks. Those tasks support some modules of the system: Data acquisition, data entry, knowledge management, inference and interface. The flexibility and efficiency of this kind of system emerges from the intelligent agents and their interaction.
- **Accuracy**: The data available in the IDSS need to be accurate and reliable. The system need to have an autonomous mechanism to a pre-validation of the data. The final validation will be always done by a Human, normally by the nurse staff. This operation should be done on

the ENR, moments after collection. With this, the user is sure that the data he can see online is guaranteed true.

- Safety: All patient data should be safely stored in the database. The data security has to be ensured the access should be restricted. This is the one of the most critical aspects in this type of approach.
- **Pervasive / Ubiquitous –** The system need to be prepared to work in ubiquitous devices like notebooks, PDAs and mobile phones. The internet plays an important role making the system available for users in anyplace. The ICU access policy should be available.
- **Privacy**: There are two types of privacy: i) related to the patient and; ii) related to the health care professional. The patient identification should be always hidden to the people out of hospital. On the other hand the pieces of information recorded on this environment need to be identified and associated to one user, in order to find out responsibilities. Both types of identifications should be protected and masked.
- **Secure Access from Exterior**: The hospital access point has to be protected from exterior connections and encrypted. A Virtual Private Network (VPN) with appropriate access protocols is a good option. Only people who have access to the ICU can see the information and operate, locally or remotely, with the IDSS. This system should implement a secure policy access and be prepared to work in a protected environment.
- **User Policy**: The IDSS should include an inside (ICU environment) and an outside (remote connections) access policy, e.g. where and who can be consult or edit the data.

DATA ACQUISITION IN ICU

KDD Process in ICU

The Figure 1 shows the data sources and the Knowledge Discovery in Database (KDD) process implemented in the ICU.

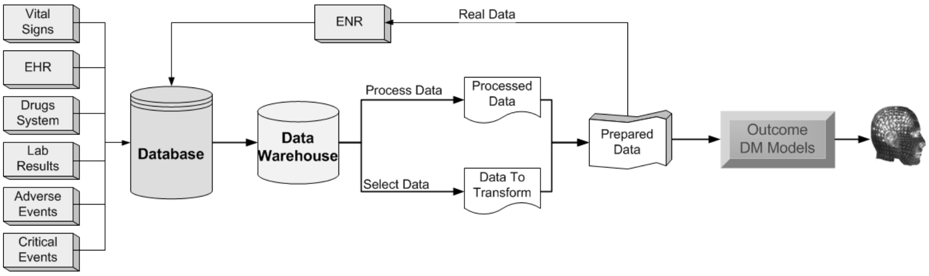

Figure 1. ICU Knowledge Discovery in Database Process

The data used for the knowledge discovery process is collected from three distinct data sources: Laboratory Results (LR), Bedside Monitors (BM), and Electronic Health Record (EHR).

After the data be received, a pre-processing agent runs in order to validate the data received, according to the limits defined by ICU (Portela et al., 2011). The data is then prepared to support the identification of critical events and to evaluate the SOFA level. At the same time the values will be classified as critical or non-critical depending on they are inside or outside the normal range (Table3).

In order to obtain the maximum number of electronic data an Electronic Nursing Record (ENR) has been developed to integrate a high number of hospital data sources like Electronic Health Process (EHR), lab results, allowing for data acquisition, data monitoring and data validation, electronically, online and in real-time. After the data is collected, these will be prepared and transformed to be used in the distributed data mining approach.

ENR delivers data to the score agent to automatically and in real-time obtain the Critical Events and SOFA results. Figure 2 presents an overview of the process.

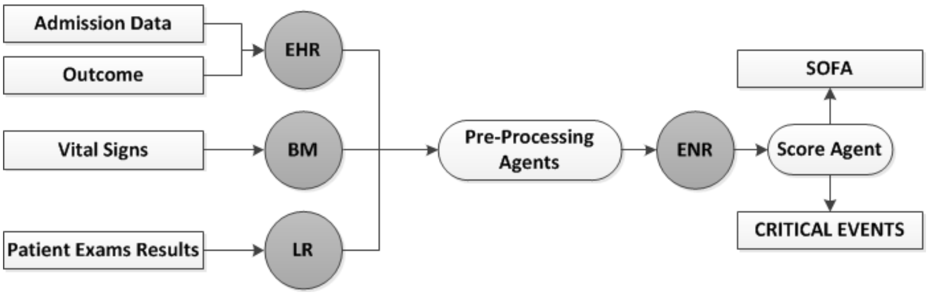

Figure 2. ICU Knowledge Discovery in Database Process

Data Set Description

The data used in this approach were collected in real-time and are related with patient who had an entire stay with a full monitoring in ICU in the first 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) (Vilas-Boas, Santos, Portela, Silva, & Rua, 2010) .

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 BP, HR, SPo2 and HR were calculated the AEC, EC and a set of ratios. Table 2 show the descriptions of each ratio and the possible values. CE was defined by a panel of experts (Á Silva, Pereira, Santos, Gomes, & Neves, 2003). If a physiological parameter is out of its normal range (Álvaro Silva, Cortez, Santos, Gomes, & Neves, 2008) 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 values that define if some value is critical or not and the max / min values that define the normal range is present in the Table 3. Other 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.

In intensive care, there are some scores to assess severity of illness, like the Sequential Organ Failure Assessment (SOFA), which is commonly used in ICU on a daily basis to score the degree of dysfunction/failure of six organic systems – Cardiovascular, Respiratory, Renal, Liver, Coagulation and Neurological (Vincent et al., 1996). SOFA is scored in a scale from 0 (normality) to 4 (failure) for each organic system. In this experiment, we transformed the SOFA scores in binary variables, where 0 describes normality and 1 describes dysfunction/failure and comprises the original SOFA.

The variables required to calculate de SOFA scores derive from heterogeneous sources: with different frequencies, as shown in Table 4.

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	Number of accumulated critical events of each VAR occurred	[0; 1]

Variable	Description	Range
/ **EC_max**	Maximum number of critical events possible in an hour	
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)

Table 3. The protocol for the out of range physiologic measurements(Álvaro Silva, et al., 2008)

	BP	SpO2	HR	UR
Normal Range	90 − 180mmHg	≥ 90%	60 − 120bp	≥ 30ml/h
Event (a)	≥ 10min.	≥ 10min.	≥ 10min.	≥ 1h
Event (b)	≥ 10min. in 30min	≥ 10min. in 30min	≥ 10min. in 30min.	-
Critical Event (a)	≥ 1	≥ 1	≥ 1	≥ 2
Critical Event (b)	≥ 1h in 2h	≥ 1h in 2h	≥ 1h in 2h	-
Critical Event (c)	< 60mmHg	< 60mmHg	< 30bpm ∨ > 180bpm	≤ 10ml/h

Where,
(a) Defined when continuously out of range;
(b) Defined when intermittently out of range;
(c) Defined anytime;

Table 4. Data sources for sofa score calculation

SOFA	Variables	Source	Frequency
Cardiovascular	Blood Pressure Dopamine, dobutamine,	BM LR	Minute Daily
Respiratory	PaO_2/FiO_2	LR	Daily
Renal	Creatinine	LR	Daily
Liver	Bilirubin	LR	Daily
Coagulation	Blood plates	LR	Daily
Neurological	Glasgow Coma Score	EHR	Hourly

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 techniques. The next sets represent the variables available, for each measure:

SOFA $_{\text{Cardio, Resp, Renal, Liver, Coagulat, neuro}}$ = {0,1}
Ratios = {ACE $_{\text{BP/hours in ICU}}$, ACE $_{\text{SO2/Hours in ICU}}$, ACE $_{\text{HR/Hours in ICU}}$, ACE $_{\text{Ur/Hour in ICU}}$}
EC = {EC $_{\text{Blood Pressure}}$, EC $_{\text{Oxygen Saturation}}$, EC $_{\text{Heart Rate}}$, EC $_{\text{Urine Output}}$}
ACE = {ACE $_{\text{Blood Pressure}}$, ACE $_{\text{Oxygen Saturation}}$, ACE $_{\text{Heart Rate}}$, ACE $_{\text{Urine Output}}$}

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 (Luo, et al., 2007). 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 (; Dam, 2008; Orriols-Puig & Bernadó-Mansilla). 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

Specific Classifier Method (SCM)

Specific Classifier Method (SCM) only preserves discrete classifiers in the global model (M. F. Santos, Mathew, & Santos). 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.

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 (M. F. Santos, et al., 2010) .If the accuracy of a classifier is greater than the cut_ off_ threshold value then that classifier will be stored in the global model.

Generalized Classifier Method (GCM)

Generalized Classifier Method (GCM) only preserves more general classifiers in the global model (M. F. Santos, et al., 2010). The main intention of the GCM is to induce a global model with the most general classifiers. The most general classifiers can represent all less general classifiers therefore in GCM. The system doesn’t allow for less general classifiers into the global model. The parameter of the more general classifier which is already in the global model is updated with the value of the less general or similar classifier. In other case, if the new classifier is more general than the classifier that are already in the global model, then all less general classifiers have to be removed from the global model and the parameter of the new classifier are updated with the parameters of all removed classifiers. The initial process of GCM is to collect all local models from the distributed sites and store them into a global model. All classifiers whose condition and action parts match to the collected local models are stored and its parameters are updated with the parameters of the other matched classifiers.

Weighted Classifier Method (WCM)

Weighted Classifier Method (WCM) only maintains the highest weighted classifiers in the global population according to the global model size (M. F. Santos, et al., 2010). The purpose of the WCM is to calculate the quality of the classifiers from its parameters and eliminate all weightless classifiers from the global model. Global model size derives from the local model size. The accuracy of the classifiers is considered as the weight of a classifier. Classifier's accuracy needs to be normalized because each local model may have a different background. Therefore, the accuracy of a classifier needs to be multiplied by the ratio between the size of the local training data set and the global training data set. Initially, the system collects and sorts all the classifiers in the local model in a descending order of the weights, then selects the classifiers that are in the range of the global population size (to crowd the population). The global population in WCM cannot represent all the classifiers in the local models because the less weighted classifiers wouldn't be included in the global population. Algorithm 3 explains the workflow of the WCM.

Model sampling Method (MSM)

Model Sampling Method (MSM) is another strategy for constructing the global model from distributed local models. The main intension of MSM is to replicate the classifiers depending on the experience of each classifier. Each time a classifier is correctly matched with an example (training data), the value of number of match of that classifier will be increased by one. Therefore the experience of a classifier is equivalent to the number of match of a classifier. The system replicates the classifier proportionally to the value of experience of a classifier. During sampling, all don't care symbols in the rule condition are replaced by other suitable values. But parameters of the replicated classifiers received the same values from the base classifiers. After sampling, replicated classifiers have to be filtered based on some quality criteria. The quality of a classifier is defined from the accuracy of that classifier. In MSM, the system will filter the classifier based on the user defined quality level.

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.

DDM Experiments

Table 5 shows the global model testing accuracies attained for the SCM, MVM, GCM, WCM and MSM 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 (table 6) MVM is the best because the global population size of the MVM is always smaller than the global population size comparatively to the other four methods. Testing accuracies increase in proportion to the population size as expected, for example, almost 72% of accuracy is achieved with local population size of 50, near to 80% of accuracy is achieved with a local population size of 100, approximately 88% of accuracy is achieved with local population size of 200, and nearly 93% of accuracy is achieved with local population size of 400. Higher population sizes were not considered in order to avoid overfitting phenomena.

Table 5. Testing accuracies attained by DDM models.

Number of Nodes	Iterations	Local Population Size	Accuracy				
			SCM	MVM	GCM	WCM	MSM
10	10,000	50	0.716 ± 0.0110	0.7132 ± 0.01252	0.7147 ± 0.0118	0.7144± 0.0125	0.7110± 0.012
10	10,000	100	0.7987 ± 0.01586	0.7987 ± 0.0175	0.7918 ± 0.0168	0.7998± 0.01555	0.7980± 0.0163
10	10,000	200	0.8784 ± 0.01715	0.876 ± 0.01511	0.8784 ± 0.0173	0.8789± 0.01518	0.88.12± 0.017
10	10,000	400	0.925 ± 0.009	0.92606 ± 0.0088	0.9261 ± 0.010	0.9256± 0.0091	0.9243± 0.0102
10	20,000	50	0.7116 ± 0.0203	0.723 ± 0.0318	0.7097 ± 0.01843	0.7165± 0.0127	0.7097± 0.0267
10	20,000	100	0.80 ± 0.0159	0.807 ± 0.0217	0.8130 ± 0.02	0.8076± 0.02075	0.8101± 0.022
10	20,000	200	0.8794 ± 0.060	0.8722 ± 0.01589	0.8776 ± 0.016	0.8724± 0.01583	0.8777± 0.01445
10	20,000	400	0.925 ± 0.0099	0.9229 ± 0.0123	0.9226 ± 0.00992	0.9225± 0.01086	0.9230± 0.0113
10	30.000	50	0.712 ± 0.018	0.7188 ± 0.0151	0.723± 0.01511	0.71919 ± .0150	0.7186± 0.01528
10	30,000	100	0.807 ± 0.0173	0.8024 ± 0.0167	0.80818± 0.016	0.80281 ± .0158	0.8024± 0.0166
10	30.000	200	0.875 ± 0.019	0.8723 ± 0.0179	0.8781 ± 0.019	0.87313 ± 0.01785	0.8743± 0.015
10	30,000	400	0.9244 ± 0.0085	0.925 ± 0.01153	0.9239 ±0.126	0.9264± 0.01112	0.9251±0.011747
20	5,000	25	0.7203 ± 0.0192	0.7345 ± 0.0232	0.7424 ± 0.01889	0.7345± 0.02323	0.7429± 0.02497
20	5,000	50	0.8028 ± 0.0176	0.797 ± 0.0177	0.8029 ± 0.0179	0.7983± 0.0154	0.7980± 0.0164
20	5,000	100	0.879 ± 0.0186	0.8781 ± 0.01084	0.8803 ± 0.107	0.87919 ± 0.0114	0.8766± 0.01164
20	5,000	200	0.932 ± 0.0130	0.927 ± 0.00674	0.9269 ± 0.0066	0.92617 ± 0.0067	0.9256± 0.0068
20	10,000	25	0.72 ± 0.018	0.721 ± 0.0158	0.7234 ± 0.016	0.7220± 0.01587	0.7262± 0.01564
20	10,000	50	0.805 ± 0.0192	0.8061 ± 0.0197	0.81± 0.0188	0.8094± 0.01514	0.8054± 0.01982
20	10,000	100	0.8824 ± 0.0167	0.884 ± 0.0151	0.8856 ± 0.01716	0.8839± 0.01518	0.88156 ±0.015
20	10,000	200	0.9298 ± 0.0153	0.9369 ± 0.0118	0.934± 0.01309	0.93717 ± 0.00914	0.9373± 0.01108
20	15,000	25	0.7197 ± 0.0965	0.7158 ± 0.0212	0.7197 ± 0.019	0.7151± 0.0217	0.7156± 0.02111
20	15,000	50	0.8091 ± 0.0129	0.8054 ± 0.0134	0.8086 ± 0.0132	0.80567 ± 0.01273	0.8052± 0.01342
20	15,000	100	0.8695 ± 0.0135	0.8699 ± 0.0132	0.8686 ± 0.0149	0.8698± 0.0131	0.87004 ±0.013

20	15,000	200	0.9325 ± 0.00977	0.9325 ± 0.01022	0.9327 ± 0.01	0.9311± 0.01086	0.93037 ±0.0107

Table 6. Global Population Sizes for DDM models.

Number of Nodes	Iterations	Local Population Size	Global Population Size				
			SCM	MVM	WCM	GCM	MSM
10	10,000	50	485.8 ± 4.87	381.3 ± 10.187	500±0	475.8± 7.39	464.77± 11.61
10	10,000	100	955 ± 5.35	655.7 ± 9.2141	1000±0	920.8± 10.992	930.66± 17.66
10	10,000	200	1884.8 ± 12.23	1070.8 ± 20.48	2000±0	1792.8± 13.9267	1824.66 ± 49.189
10	10,000	400	3730.9 ± 17.615	1710.7 ± 33.40	4000±0	3490± 25.490	3626.33 ± 71.4923
10	20,000	50	486.4 ± 3.687	383.2 ± 9.635	500±0	476.3± 4.243	466.88± 10.782
10	20,000	100	958.8 ± 7.08	648.2 ± 11.698	1000±0	919.11± 6.7404	911± 29.417
10	20,000	200	1885 ± 11.72	1067.5 ± 21.36	2000±0	1794.5± 12.3939	1806± 47.132
10	20,000	400	3724 ± 12.18	1713 ± 42.62	4000±0	3471.4± 29.7814	3650.77 ± 109.529
10	30.000	50	484 ± 2.366	382.5 ± 12.020	500±0	474.8± 3.224	466.4± 16.2699
10	30,000	100	958.7 ± 4.80	654.8 ± 10.695	1000±0	928.1± 7.54	907.2± 24..0268
10	30.000	200	1890.2 ± 9.96	1063. ± 31.287	2000±0	1793.6± 18.9103	1801± 40.032
10	30,000	400	3720.1 ± 20.82	1705.5 ± 24.24	4000±0	3477.2± 18.0542	3713± 117.5859
20	5,000	25	488.2 ± 3.119	394.1 ± 6.789	500±0	477.1± 5.4863	474.77± 11.1200
20	5,000	50	959.1 ± 6.55	676.1 ± 17.47	1000±0	933.5± 7.8634	898.66± 58.1133
20	5,000	100	1890 ± 11.2570	1111.9 ± 28.68	2000±0	3497.9± 23.7133	1823± 32.3254
20	5,000	200	3733 ± 14.2126	1779.7 ± 31.16	4000±0	1807.4± 15.2257	3792± 0.85.8616
20	10,000	25	486 ± 4.13	391.7 ± 6.412	500±0	476± 5.37	7232± 0.0156
20	10,000	50	962.6 ± 4.501	669.3 ± 16.97	1000±0	932.8± 7.13	914.11± 22.0615
20	10,000	100	1892 ± 9.04	1101.8 ± 16.87	2000±0	1802.9± 12.087	1818.8± 25.1354
20	10,000	200	3729.9 ± 13.194	1757.3 ± 33.50	4000±0	3493.8± 16.565	3709.22 ± 55.3655
20	15,000	25	486.6 ± 4.5509	389.6 ± 7.29	500±0	476.9± 6.9033	466.2± 10.9625
20	15,000	50	961.6 ±	673.2 ±	1000±0	931.7±	934.2±

			7.381	38.473		11.10	22.8609
20	15,000	100	1886 ± 10.286	1110.7 ± 10.69	2000±0	1804.2± 13.18922	1835.9 ± 21.9820
20	15,000	200	3738.8 ± 20.339	1777.6 ± 019.18	4000±0	3510± 28.386	3739.8± 69.517

CDM Experiments

The testing accuracies obtained by the CDM approach are presented in Table 7. They 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 proportionally to the population size. For each experiment (CDM1, CDM12) the corresponding DDM tests are identified.

Table 7. Testing accuracies for the CDM method.

Iteration	Population Size	Accuracy	DDM
100,000	500	0.56232 ± .17046	DDM1, DDM13
100,000	1000	0.6035 ± 0.182586	DDM2, DDM14
100,000	2000	0.6585 ± 0.1992	DDM3, DDM15
100,000	4000	0.7086 ± 0.2138	DDM4, DDM16
200,000	500	0.565 ± 0.170825	DDM5, DDM17
200,000	1000	0.5974 ± 0.1808	DDM6, DDM18
200,000	2000	0.64885 ± 0.1962	DDM7, DDM19
200,000	4000	0.7114 ± 0.2146	DDM8, DDM20
300,000	500	0.5585 ± 0.1689	DDM9, DDM21
300,000	1000	0.5996 ± 0.1814	DDM10, DDM22
300,000	2000	0.6507 ± 0.1965	DDM11, DDM23
300,000	4000	0.7156 ± 0.216	DDM12, DDM24

DISCUSSION AND RELATED WORK

The main goal of this work was to induce global data mining models and compare the performance of CDM versus the DDM methods applied to predict the outcome of patients in ICU environments. Five strategies described above were developed and tested in order to construct the global model from a set of distributed local models. The global model in the CDM method is obviously representing the overall problem (dataset) in the distributed sites because that model is generated from the global data without any intervention. Though table 5 and 6 show that DDM attained better accuracies for similar settings. Another advantage assignable to the DDM approach is that it avoids sending large size of data from different sites to a central site. DDM data is processed at each distributed sites and generate learning models. As mentioned in the introduction, the size of the training data is always very large than the data mining model size (classifiers population) and the computational and communicational times associated to DDM tend to be very much lower the required for CDM. This way of processing has two main advantages: 1) privacy of the data; and 2) less communication costs (Schmidhuber, 2003).

Among the DDM strategies, MVM needs smaller populations of classifiers to attain similar accuracies. This is very important because smaller models are preferable in domains like the ICU ones, where the real time is a requirement so the computational time spent to run the models is critical. When compared to the models induced in each node the global models perform better.

It should be stressed that those strategies are not based on any specific domain. The main idea behind these different strategies is to understand the behaviour of a global model constructed with classifiers copied from the local modes. The first two strategies (SCM and GCM) shape the global model based on the rules (Condition and Action of the classifier), next two strategies (MVM and WCM) shape the global model based on the classifiers' parameter values. The last strategy, the MSM, shapes the global model based on the replication function. Just to have an idea, the MVM approach attains an accuracy of 0.932± .0102 with a population of 1777.6± 19.18 classifiers. The CDM approach needs to induce a model with 4000 classifiers and attains an accuracy of 0.7156± 0.216 (the best value for CDM).

Considerable related work could be found in parallel and distributed implementations of LCS. The experimental work is mainly oriented to compare the speed-up attained. Our work points to a different direction. We are primarily concerned with the induction of global models based on local models. Similarities can be established with meta-learning approaches. The goal of the meta-learning is to construct the global population of classifiers from a collection of inherently distributed data sources (Cesario, Congiusta, Talia, & Trunfio, 2008). GALE (Genetic and Artificial Life Environment) is another related work in the distributed data mining area. GALE is a fine grained parallel genetic algorithm based on a classification system (Llora & Garrell, 2001). Learning classifier system ensembles with rule sharing is another associated work relating to in the parallel and distributed LCS (Bull, Studley, Bagnall, & Whittley, 2007).

The future use of this strategy in ICUs is very attractive because:

- Enables the use of data collected from geographically distinct sites. Those sites can belong to same hospital unit or to different units;
- Allow for knowledge merging. The global models are induced from different sub models capturing specific and general trends. This means that the empirical knowledge existing in different sites can be shared with the other sites. The system is able to share knowledge like

the professionals do when they participate in workshops to share experiences.

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 achieved similar testing accuracies but the global population size of MVM is smaller than the global population size of the other approaches. The results are very important in areas were distributed data should be considered without discharging the local models induction as is the ICU. The approach will enable in the near future the share of local knowledge by the other sites.

Further work will include the application of DDM to the prediction of organ failure/dysfunction.

ACKNOWLEDGMENTS: The authors would like to express their gratitude to FCT (Foundation of Science and Technology, Portugal), for the financial support through the contract INTCare - PTDC/EIA/72819/2006. The work of Filipe Portela was supported by the grant SFRH/BD/70156/2010 from FCT.

REFERENCES

Bull, L., Studley, M., Bagnall, A., & Whittley, I. (2007). Learning classifier system ensembles with rule-sharing. *Evolutionary Computation, IEEE Transactions on, 11*(4), 496-502.

Cesario, E., Congiusta, A., Talia, D., & Trunfio, P. (2008). Data analysis services in the Knowledge Grid. *Data mining techniques in grid computing environments*, 17-36.

Dam, H. H. (2008). A scalable evolutionary learning classifier system for knowledge discovery in stream data mining.

Das, S. K., & Roy, N. (2008). Learning, Prediction and Mediation of Context Uncertainty in Smart Pervasive Environments. In R. Meersman, Z. Tari & P. Herrero (Eds.), *On the Move to Meaningful Internet Systems: Otm 2008 Workshops* (Vol. 5333, pp. 820-829). Berlin: Springer-Verlag Berlin.

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

Gain, G. (2006). Grid Gain - key features. Retrieved 8-2-2011, 2011, from http://www.gridgain.com/key_features.html

Giani, A., Starita, A., & Vanneschi, M. (1999). Parallel cooperative classifier systems. *These de doctorat de l'université de Pise, Italie*.

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

Le Gall, J. R., Lemeshow, S., & Saulnier, F. (1993). A new Simplified Acute Physiology Score (SAPS II) based on a European/North American multicenter study. *JAMA, 270*(24), 2957-2963.

Llora, X., & Garrell, J. M. (2001). *Knowledge-independent data mining with fine-grained parallel evolutionary algorithms*.

Luo, J., Wang, M., Hu, J., & Shi, Z. (2007). Distributed data mining on agent grid: issues, platform and development toolkit. *Future Generation Computer Systems, 23*(1), 61-68.

M.Cannataro, A. C., A. Pugliese, D.Talia, P. Trunfio. (2004). Distributed Data Mining on Grid: Services, Tools, and Applications. *IEEE TRANSACTIONS ON SYSTEM, MAN, AND CYBERNETICS- PART B: CYBETNETICS, 34*(6).

Orriols-Puig, A., & Bernadó-Mansilla, E. (2006). *A further look at UCS classifier system*.

Portela, F., Gago, P., Santos, M. F., Silva, A., Rua, F., Machado, J., et al. (2011). *Knowledge Discovery for Pervasive and Real-Time Intelligent Decision Support in Intensive Care Medicine*. Paper presented at the *KMIS* 2011- International Conference on Knowledge Management and Information Sharing.

Portela, F., Santos, M. F., Silva, Á., Machado, J., & Abelha, A. (2011). *Enabling a Pervasive approach for Intelligent Decision Support in Critical Health Care*. Paper presented at the *HCist 2011* – International Workshop on Health and Social Care Information Systems and Technologies.

Santos, M. F. (1999). *Learning Classifier System in Distributed environments.* University of Minho.

Santos, M. F., Mathew, W., Kovacs, T., & Santos, H. (2009). A grid data mining architecture for learning classifier systems. *WSEAS Transactions on Computers, 8*(5), 820-830.

Santos, M. F., Mathew, W., & Santos, H. (2010). *GridClass: Strategies for Global Vs Centralized Model Construction in Grid Data Mining*.

Santos, M. F., Mathew, W., & Santos, H. D. (2011). *Grid data mining by means of learning classifier systems and distributed model induction*.

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

Schmidhuber, J. (2003). INCREMENTAL LEARNING.   Retrieved 01-07-2010, 2010, from http://www.idsia.ch/~juergen/icmlkolmogorov/node9.html

Silva, Á. (2007). *Modelos de Inteligência Artificial na análise da monitorização de eventos clínicos adversos, Disfunção/Falência de órgãos e prognóstico do doente critico.* Universidade do Porto, Porto.

Silva, Á., Cortez, P., Santos, M. F., Gomes, L., & Neves, J. (2008). Rating organ failure via adverse events using data mining in the intensive care unit. *Artificial Intelligence in Medicine, 43*(3), 179-193.

Silva, Á., Pereira, J., Santos, M., Gomes, L., & Neves, J. (2003). *Organ failure prediction based on clinical adverse events: a cluster model approach*, 3th International Conference on Artificial Intelligence and Applications.

Suter, P., Armaganidis, A., Beaufils, F., Bonfill, X., Burchardi, H., Cook, D., et al. (1994). Predicting outcome in ICU patients. *Intensive Care Medicine, 20*(5), 390-397.

Varshney, U. (2009). *Pervasive Healthcare Computing: EMR/EHR, Wireless and Health Monitoring*: Springer-Verlag New York Inc.

Vilas-Boas, M., Santos, M. F., Portela, F., Silva, Á., & Rua, F. (2010). *Hourly prediction of organ failure and outcome in intensive care based on data mining techniques*. Paper presented at the 12th International Conference on Enterprise Information Systems.

Vincent, J. L., Moreno, R., Takala, J., Willatts, S., De Mendonca, A., Bruining, H., et al. (1996). The SOFA (Sepsis-related Organ Failure Assessment) score to describe organ dysfunction/failure. *Intensive care medicine, 22*(7), 707-710.