A Clustering-Based Multi-Layer Distributed Ensemble for Neurological Diagnostics in Cloud Services

Morshed Chowdhury, Jemal Abawajy, *Senior Member, IEEE,* Andrei Kelarev and Herbert F. Jelinek, *Member, IEEE*

Abstract—This paper investigates the problem of minimizing data transfer between different data centers of the cloud during the neurological diagnostics of cardiac autonomic neuropathy (CAN). This problem has never been considered in the literature before. All classifiers considered for the diagnostics of CAN previously assume complete access to all data, which would lead to enormous burden of data transfer during training if such classifiers were deployed in the cloud. We introduce a new model of clustering-based multi-layer distributed ensembles (CBMLDE). It is designed to eliminate the need to transfer data between different data centers for training of the classifiers. We conducted experiments utilizing a dataset derived from an extensive DiScRi database. Our comprehensive tests have determined the best combinations of options for setting up CBMLDE classifiers. The results demonstrate that CBMLDE classifiers not only completely eliminate the need in patient data transfer, but also have significantly outperformed all base classifiers and simpler counterpart models in all cloud frameworks.

Index Terms—cardiac autonomic neuropathy, distributed ensembles, classifiers, cloud services

1 Introduction

Cloud computing offers cost-effective options of information technology services to health care via computer networks. It is widely adopted because of its many advantages including cost-effectiveness, accessibility and scalability [1]. Cloud storage allows patients, general practitioners, and hospital data to be accessed via computer networks at any time and from anywhere using mobile devices [2]. The use of electronic health record has increased the amount of health care data to be stored and processed in the cloud. Software as a service is also offered to healthcare by the cloud, which leads to the challenges of optimizing the operation of virtual machines and data center networks [3], [4]. In the development of new algorithms it is now essential to take into account the requirements of reducing processing and communication costs associated with workflow management in the cloud [5]. It is also essential to plan the distribution of workload in the cloud data center network for the reduction of energy consumption [6].

Cloud services can offer considerable advantages for personalized medicine and participatory health care, which lead to precise and individualized approaches for the prevention, diagnosis, and therapy of patients [7], [8] and involve real-time collection, monitoring and interpretation of data from wearable and environmental sensors [9] as well as processing general practice and hospital data via smartphones and tablets [10]. The cost-effective and flexible solutions make personalized and participatory health care more accessible for patients not only in metropolitan areas but also in rural and remote regions. New methods need to be developed to facilitate the use of the cloud in the personalized and participatory medicine.

This paper is devoted to new algorithms for the neurological diagnostics of cardiac autonomic neuropathy (CAN), which is a cardiac condition quite common in diabetes patients. To minimize data transfer between different data centers and improve the accuracy of the neurological diagnostics of CAN in the cloud, we introduce a new model of clustering-based multi-layer distributed ensemble (CBMLDE). The proposed method combines machine learning and cloud technology to allow processing of large ambulatory electrocardiography data by eliminating patient data transfer between nodes of the cloud.

We used an extensive database created by the Diabetes Complications Screening Research Initiative (DiScRi) at Charles Sturt University. Section 3 contains background information on DiScRi and the groups of features for different data centers of the cloud. This is the first article concerned with algorithms for the processing of DiScRi data in the cloud.

In [11] the authors explored several ensembles

- *Morshed Chowdhury is with the Parallel and Distributed Computing (PARADISE) Lab, Deakin University, Geelong, Australia (email:muc@deakin.edu.au).*
- *Jemal Abawajy is with the PARADISE Lab, Deakin University, Geelong, Australia (email: jemal@deakin.edu.au).*
- *Andrei Kelarev (corresponding author) is with the PARADISE Lab, Deakin University, Geelong, Australia (email: andreikelarev-deakinuniversity@yahoo.com).*
- *Herbert F. Jelinek is with Charles Sturt University, Albury, NSW, Australia (email:HJelinek@csu.edu.au).*

of decision trees for the neurological diagnostics of CAN. Clinical data associated with diabetes more often than not is obtained from several sources such as pathology laboratories for blood tests, cardiology, ophthalmology, podiatry/vascular medicine, endocrinology/diabetology clinics and the general practice presenting a natural multi-node paradigm. However, larger data sets from diverse locations require new theoretical models and new computational experiments are required to develop ensemble classifiers for work in the cloud. The distributed nature of data stored in the cloud creates different requirements for the operation of an ensemble classifier. An ensemble classifier allocated in one node of the cloud, or in one location of the cloud service provider, cannot efficiently process data available at another node, since this would lead to transfer of large amounts of data between different nodes. It is essential to minimize the transfer of data between different nodes of a distributed database and to investigate ways of transferring aggregated or compressed portions of data between nodes so that data stored at one node will be able to contribute to the training of a classifier allocated at another node.

Our CBMLDE classifiers use a clustering-based selection strategy, which serves to reduce the number of base classifiers in the ensemble and at the same time to improve its performance. Ensembles incorporating selection strategies have never been applied to the classification of CAN. This method is important for cloud applications, since reduction of size makes it easier to manage the algorithm in the cloud.

This is the first article that investigates distributed ensembles for the neurological diagnostics of cardiac autonomic neuropathy (CAN). We introduce a new model of CBMLDE classifiers designed to minimize data transfer between different data centers of the cloud. We use data from the DiScRi database to conduct a comprehensive set of experiments determining the best options to be implemented in the CBMLDE classifier for the neurological diagnostics of CAN in the cloud.

The novel character of our model of a CBMLDE classifier is in its multi-layer structure designed to minimize data transfer between different data centers and at the same time to achieve high accuracy of the neurological diagnostics of CAN. Following [3], [12], here we use the term "multi-layer" to refer to the structure of the system and to emphasize the difference in comparison with other studies that looked at multiple levels, stages or steps of operation of the corresponding systems, as for example, in [13], [14], [15] and [16]. Distributed ensembles with multiple layers in their structure have not been considered in the literature. In particular, they have never been applied for the neurological diagnostics of CAN, nor as a model applied in cloud computing.

The novelty of these classifiers is in their multi-layer structure with four layers combined with novel selection strategies. Our selection strategies are based on clustering, since this method has proved useful in many studies. Clustering was used in conjunction with selection strategies, for instance, in [17], [18]. Clustering techniques have been applied for solving many problems in biomedical informatics and health informatics, for example, in [13], [19], [20], [21], [22], [23]. More explanations of the structure and operation of CBMLDE classifiers are given in Section 2.

The innovation of this work is in comprehensive investigation of the new model of CBMLDE classifiers designed to minimize data transfer between different data centers of the cloud and to enhance the diagnostic accuracy for CAN within a cloud environment. This is the first paper concentrating on the investigation of the accuracy of the neurological diagnostics of CAN taking into account the requirement to minimize data transfer between different data centers in the cloud.

This article presents a systematic set of experiments assessing the performance of novel CBMLDE classifiers, outlined in Section 2. More details on the DiScRi database and the groups of data for different data centers are presented in Section 3. Our experiments compare the efficiencies of various combinations of base classifiers and ensembles to be included in CBMLDE classifiers. The outcomes of these experiments and are explained in Section 6. The conclusions are given in Section 7.

2 The Structure and Operation of CBMLDE Classifiers

The major problem that has to be overcome for the successful operation of a diagnostic classifier in the cloud in the reduction of data transfer required for training of the classifier. Once a diagnostic classifier has been trained, to diagnose any new patient it is very easy to transfer all data of one patient to one of the cloud nodes where the model of classifier has been allocated for the future operation. However, the process of initial training of the classifier requires all training data of all patients to be transferred to one Virtual Machine (VM) at the cloud node where the classifier can be trained. With the introduction and broad use of electronic health records the patient data will be stored in several different data centers, because each Cloud Service Provider (CSP) operating for large client base in health services will have to maintain several data centers, and also because different medical specialists may be allowed to choose their CSPs. However, to train a classifier, all training data will have to be transferred to one Virtual Machine, where the classifier is to be trained. Data transfer of such scale is infeasible, for example, in situations where it is desirable to use all data of patients in a large

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2016.2567389, IEEE Transactions on Cloud Computing

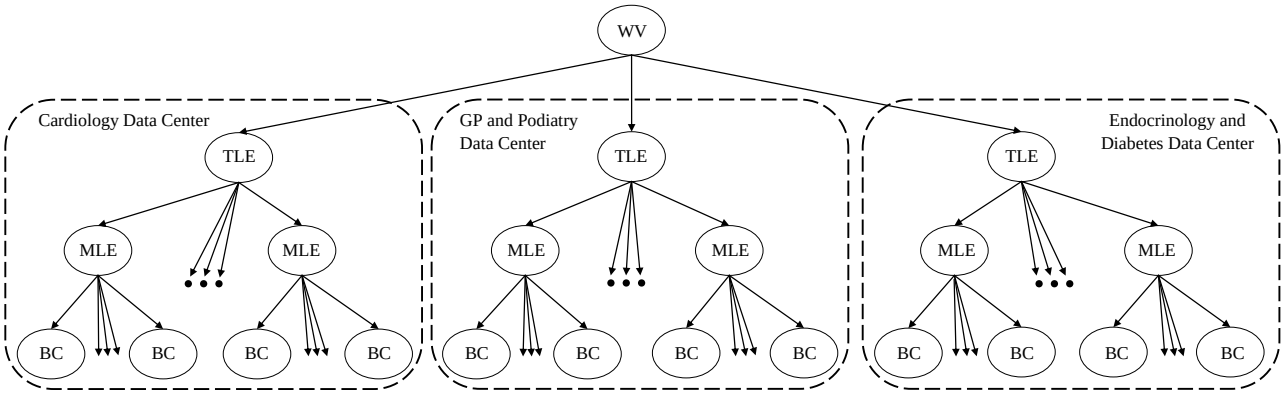

Fig. 1. The multi-layer structure of a CBMLDE classifier with three data centers.

geographic area or the whole country to achieve high accuracy of training.

Here we introduce a new model of the Clustering-Based Multi-Layer Distributed Ensemble that completely eliminates the need to transfer data during training of the diagnostic classifier. To this end the model uses a multi-layer structure. The structure of every CBMLDE classifier has four layers depicted in Figure 1 for the case of three data centers and in Figure 2 for the case of six data centers. The following notation is used in these diagrams.

- Base classifiers (BC) are deployed at the bottom layer. They process features from instances of data in the data set and pass on their predictions to their parent ensembles on the next layer.
- Middle layer ensembles (MLE) are deployed at the third layer from the top. They process the outcomes of the BCs and pass on the outcomes to their parent ensembles on the higher layer.
- Top layer ensembles (TLE) are deployed at the second layer from the top. They process the outcomes supplied by the MLE and send their predictions to the top layer.
- Weighted vote (WV) is deployed at the top layer to combine predictions of the TLE.

It uses the WV to combine the diagnostic predictions of several classifiers each of which is trained using only data allocated at the same node of the cloud where the classifier is trained. The weights for the Weighted Vote are chosen equal to the diagnostic accuracy achieved by each local classifier that is being combined, so that more accurate local classifiers contribute more towards the final diagnosis. The advantage of using the WV is that it does not need to be trained, and there is no need to transfer any data between different nodes for training the model. This means that the CBMLDE classifiers completely eliminates the overhead of data transfer for training of the classifier. Our application of the WV originally was inspired by the way vote was used to improve classifications of multiple classifiers, for example, in [24].

This is the very first article proposing a model of a distributed classifier for the diagnosis of CAN in the cloud. Distributed ensembles with multi-layer structure of this sort have not been considered in the literature before. Originally the motivation and inspiration for our study came from many different multi-layer systems (cf. [3], [12], [25]) and multi-step and multi-stage procedures (cf. [13], [14], [16], [24], [26]).

Models of the CBMLDE classifiers can be easily generated in WEKA. The generation of every CBMLDE classifier is concluded by applying a selection strategy to reduce the size of the whole ensemble and further increase its effectiveness by keeping only the most relevant base classifiers. Our selection strategy is based on hierarchical clustering produced by the Average Link heuristic, ALC [27] and Simple K-Means available in WEKA [28]. The ALP is applied to create a stable clustering combining several clusterings created by Simple K-Means.

To apply Simple K-Means and ALC to the base classifiers of a CBMLDE classifier, we need to represent these base classifiers in a vector space model. To this end we made a random selection of 60 patients $p_1, p_2, \ldots, p_{60}$ from the DiScRi database and fixed them as patients corresponding to the components of vectors to be created in order to compute the vector representations of the base classifiers. Let the number of all base classifiers in the CBMLDE ensemble be equal to K, and let the set of all base classifiers be

$$\{c_i \mid i = 1, 2, \ldots, K\}. \tag{1}$$

Then each base classifier c_i acquires the vector representation

$$\overrightarrow{w_i} = (w_{i,1}, w_{i,2}, \ldots, w_{i,60}), \tag{2}$$

where the component $w_{i,j}$ is equal to 1 if the classifier c_i predicts that the patient p_j has CAN, and $w_{i,j}$ is equal to 0 if c_i predicts that the p_j has no CAN.

Simple K-Means is the WEKA implementation of the classical k-mean clustering algorithm described

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2016.2567389, IEEE Transactions on Cloud Computing

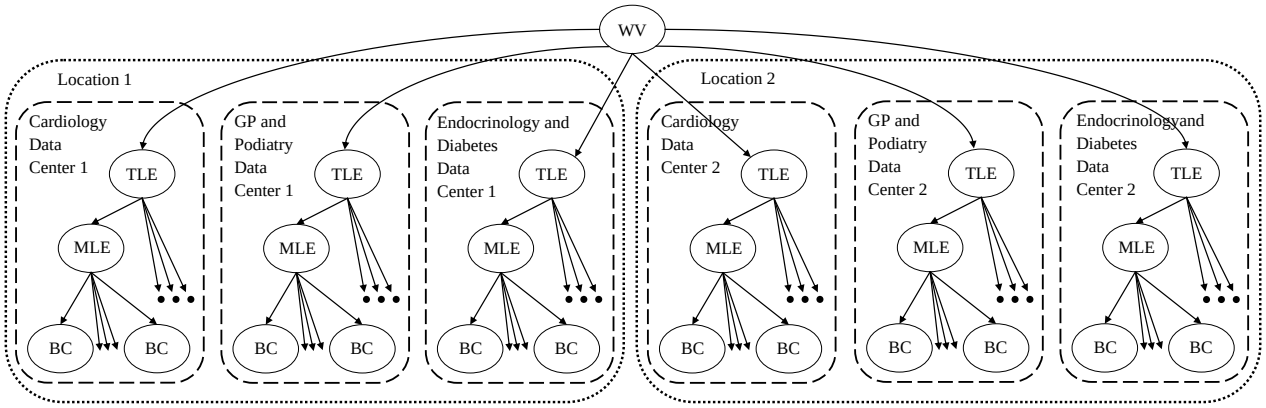

Fig. 2. The multi-layer structure of a CBMLDE classifier with six data centers.

in [29]. This algorithm randomly chooses k selects some vectors representing the base classifiers as the centroids of clusters at the initialization stage. The number of centroids is equal to the required number of clusters. Every other vector of the base classifier is allocated to the cluster of its nearest centroid. Then each iteration finds new centroids of all current clusters as a mean of all members of the cluster. This is equivalent to finding the vector such that the sum of all distances from the new centroid to all other vectors of the base classifiers in the cluster is minimal. The algorithm then reallocates all vectors of the base classifiers to the clusters of their nearest centroids. The algorithm continues these iterations until the centroids stabilize. We used SimpleKMeans with the default Euclidean distance in the vector space. We refer to [29], [30], [31] for more explanations and further references pertaining to these clustering algorithms.

The output of Simple K-Means in WEKA depends on the number of the required clusters and the value of the seed parameter. The number K of clusters in our experiments was determined by the number of base classifiers required to be selected during the selection stage. Given the number of clusters, in order to eliminate the dependence on the seed parameter, we ran Simple K-Means with ten random values of the seed, and then applied the ALC heuristic to combine these ten clusterings of the vectors of base classifiers into one stable clustering. Let us briefly discuss how the ALC operates. Denote by C_1, $C_2, \ldots, C_{10}$ the set of clusterings produced by the Simple K-Means for ten random values of the seed. To find a proposed stable combined clustering C, the ALC aims to minimizes the sum $\sum_{i=1}^{10} |\Delta(C, C_i)|$, where $\Delta(C, C_i)$ is the symmetric difference between C and C_i, and $|\Delta(C, C_i)|$ denotes the cardinality of this symmetric difference. The ALC begins the search for the optimal clustering C by considering a partition where every cluster consists of exactly one element, and each element is clustered separately. Then the ALC finds the two clusters with the smallest average distance between pairs of vectors of base classifiers in the clusters and merges them together. This merging process continues until the required number of clusters remain in the partitioning. In [27], this algorithm was implemented with running time $O(B^{(K+\log B)})$, where K is the required number of clusters, and B is the total number of all base classifiers initially generated for the whole CBMLDE classifier.

Our selection strategy for the CBMLDE classifier is based on the clustering obtained as explained above and it chooses medoids in each cluster. The *medoid* of a cluster is an instance of the dataset that either coincides with the centroid or is a nearest neighbour of the centroid of the cluster. It is very easy to compute the centroid of a cluster, since all its components are equal to the mean values of the corresponding components of all instances in the cluster. The centroid might not be an element of the dataset, and so the medoid has to be used instead.

After the selection stage is complete, the classifier is ready to be applied to the test set and new instances of data. During classification each base classifier assesses instances of data and passes their evaluation to its parent ensemble at the middle tier. Then every ensemble of the middle tier sends their combined information to the ensemble at the top tier, which outputs the final conclusion of the CBMLDE classifier.

3 Groups of Features for Data Centers in DiScRi Database

In order to investigate the data mining algorithms for the neurological diagnostics of CAN, we used a large database of test results and health-related parameters collected at the Diabetes Complications Screening Research Initiative (DiScRi) organized at Charles Sturt University [32]. The database contains over 200 features for each participant.

CAN is a condition associated with damage to the autonomic nervous system innervating the heart

TABLE 1
Examples of cardiology features in the DiScRi.

Notation	Simplified explanation of Feature
CVD Status	Does the patient have cardiovascular disease or not?
CVD % risk 5 years	Risk of CVD in 5 years determined by the doctor from other tests.
CVD years	How many years has the patient been diagnosed with cardiovascular disease?
Family History CVD	Family history of cardiovascular disease.
Angina	Is there pain in the chest present or not?
Heart Failure	Is advanced heart disease present or not?
Heart Atrial Fibrilation	Is heart rhythmm disturbance present or not?
Palpitations	Feelings of pounding heart.
Pacemaker	Device that regulates rhythm of heart.
Heart Attack	Did the patient suffer a heart attack previously?
ECG interpretation 10sec	Automated interpretation of ECG recording.
Grade 10sec	Grade for interpretation 1a, 1b are normal, 2a not so good, 2b needs to see doctor and 3 – definitely must see doctor.
PQ 10sec	PQ is interval on ECG – time for electrical conduction between point P and Q on 10 second recording.
QRS 10sec	The width of the QRS interval in milliseconds - long interval over 100msec is abnormal and shows bad conduction through ventricle.
QTc 10sec	Corrected QT interval.
QTd 10sec	QT dispersion, i.e., interval differences between recording leads.
QRS axis (degree) 10sec	Axis of QRS shows abnormal conduction of electrical impulses.

[33], [34], [35]. Automated early diagnostics of CAN is important, because it has implications for planning of timely treatment, which can contribute to an improved well-being of the patients and potentially can reduce the morbidity and mortality associated with cardiac arrhythmias in diabetes. The Ewing tests required for identification of CAN rely on assessing responses in heart rate and blood pressure to various activities, usually consisting of tests described by [33], [34]. They are the lying to standing heart rate change (LSHR), deep breathing heart rate change (DBHR), valsalva manoeuvre heart rate change (VAHR), hand grip blood pressure change (HGBP), and lying to standing blood pressure change (LSBP). In addition to these Ewing tests, the DiScRi database contains four major groups of features in the database: the cardiovascular features, endocrinology and diabetes features, podiatry features and features that are commonly collected by the general practitioners (GP). We include four tables with examples of features in these four groups, the notation used for these features in the DiScRi database, and succinct explanation of their meaning. Table 1 supplies examples of cardiology features. These are often collected for patients with various heart maladies. Examples of the general practice features and podiatry features are furnished by Tables 2 and 3, respectively. Examples of endocrinology and diabetes features are given in Table 4.

TABLE 2
Examples of GP features in the DiScRi.

Notation	Feature
Patient Age	Age of the patient.
Gender	Gender of the patient.
Waist Circumference	Waist circumference of the patient.
BMI	Body mass index is determined from height and weight (weight/height*height).
Diet	Is patient on a special diet for health reasons.
Smoking	Yes or no?
Alcohol	Yes or no?
Exercise Duration	How many hours per week does the patient exercise?
Exercise Intensity	Low/moderate/high of weekly exercises.
TC(mmol/L)	Total cholesterol in blood.
Triglyceride(mmol/L)	The level of triglyceride in blood.
HDL(mmol/L)	High density lipoprotein in blood.
LDL(mmol/L)	Low density lipoprotein in blood.
TC/HDL ratio	Ratio of total cholesterol to high density lipoprotein.
SBP average	Systolic blood pressure.
Lying SBP average	Systolic blood pressure measured for lying patient.
DBP average	Diastolic blood pressure.
Lying DBP average	Diastolic blood pressure measured for lying patient.
HT Status	Hypertension status. Does the patient have high blood pressure?

We investigate the original classification of CAN with two classes [33], [34]. For experiments presented in this paper, the values of the class variable for the diagnosis of CAN are denoted by 'no CAN' and 'CAN'. All patients without CAN have the value 'no CAN' as the class variable of their instances of data, and all patients diagnosed with an early, definite, severe or atypical levels of CAN are included in the second class denoted by 'CAN'. The class value 'no CAN' is also denoted as 'normal' in the literature [33], [34].

The collection of Ewing tests often remains incomplete due to tests being contra-indicated for patients with other aggravating conditions. Unavailable Ewing features are quite common, because many patients cannot perform some of the tests [36]. It is not always possible for the patient to undertake all of the Ewing tests. For example, some patients may be unable to perform the lying to standing tests done due to mobility challenges. Likewise, the hand grip test may

TABLE 3
Examples of podiatry features in the DiScRi.

Notation	Feature
Foot peripheral vascular disease	Is blood flow to foot impaired.
Muscle L foot	Is there muscle tension in left foot?
Muscle R foot	Is there muscle tension in right foot?
Ulcers L Leg	Are there ulcers or wounds on left leg?
Ulcers R Leg	same as 'Ulcers R Foot', but for right leg.
Reflex ankle L left	Hit patient with reflex hammer on ankle to determine nervous system function in lower limb.
Reflex ankle R leg	Same as 'Reflex ankle R foot', but for right leg.
Reflex knee L leg	Hit patient with reflex hammer either on knee to determine nervous system function in lower limb.
Reflex knee R leg	same as 'Reflex knee R foot', but for right leg.
Vibration L foot	Can the patient tell if tuning fork is placed on foot when eyes closed. Shows sensory nervous system function.
Vibration R foot	Same as 'Vibration L foot', but for right foot.
Monofilament L leg	Sensory test of lower limb - can patient feel light touch
Monofilament R leg	Same as 'Monofilament L leg', but for right leg.
L ABPI	Ankle brachial pressure index is determined by blood pressure in ankle divided by blood pressure in arm indicates vascular disease.
R ABPI	Same as 'L ABPI', but for right leg.
ABPI average	The average of 'L ABPI' and 'R ABPI'
L carville score	Special score associated with monofilament test.
R carville score	Same as 'R carville score', but for right leg.

TABLE 4
Examples of endocrinology and diabetes features in the DiScRi.

Notation	Feature
Diabetic Status	Has the patient been diagnosed with diabetes type 1 or diabetes type 2?
Diagnostic DM	How many years since the diagnosis of diabetes.
Family History DM	Family history of diabetes.
Kidney Problem	Has the patient been diagnosed with kidney problems?
Bladder Problem	Has the patient been diagnoses with bloodier problems?
Screening glucose	Level of glucose in blood after fasting.
CRP	C reactive protein – biomarker for inflammation associated with heart disease.
Hcy(mmol/L)	Homosyteine - damage to blood vessel lining also associated with atherosclerosis.
HbA1c (%)	Biomarker indicating how good blood sugar levels are controled over time.
met Hb (%)	An antioxidant in the blood.
GSH	Glutathione is an antioxidant in body - fights free radicals that damage blood vessels and nerves.
MDA	Melondealdehyde indicates change in cholesterol and possible atherosclerosis.
C5a	It is associated with clotting of material in blood vessel.
D-DIMER	It is associated with unclotting of material in blood vessel.
GFR	Glomerular filtration rate indicates kidney dysfunction.

be difficult to do due to arthritis. Some patients have ailments where forceful breathing for the Valsalva manoeuvre is undesirable. These issues often result in CAN risk assessments being made in practice on the basis of only a subset of the Ewing tests ([33], [34], [36]), but with the help of some other alternative and possibly already available features.

4 Base Classifiers for CBMLDE Classifiers

This section contains concise preliminaries on the standard base classifiers used as building blocks for constructing CBMLDE classifiers in our experiments: BayesNet, CHIRP, ConjunctiveRule, IBk, Random Tree, Ridor. These classifiers were chosen, because they are well known, represent several important classes of base classifiers, can be combined in multi-layer ensembles, are small and convenient for cloud deployment, and all of them are available in the open source WEKA [28].

BayesNet classifier learns a network structure and probability distributions for each node of a Bayesian network. It handles only discrete variables. We used the WEKA filter weka.filters.unsupervised.attribute.Discretize to discretize continuous variables of DiScRi database for BayesNet.

CHIRP is a base classifier in WEKA, which uses a set cover on iterated random projections [29]. It applies an iterative sequence, where each of the stages consists of projecting, binning, and covering. This method was designed to deal with the difficulties caused by the dimensionality of data, the computational complexity, and nonlinear separability of data.

ConjunctiveRule is a base classifier in WEKA constructing a single conjunctive rule learner that can produce predictions for numeric and nominal class variables [29]. A rule consists of antecedents and a consequent. The antecedents are combined together via logical conjunction, and the consequent is the class value for the classification. If a new test instance is not covered by this rule, then its prediction uses the default class value of the instances of the training data not covered by the rule. ConjunctiveRule selects

antecedents by computing the Information Gain of all features. It optimizes the generated rule to reduce error rate and the number of antecedents.

IBk is a WEKA implementation of the classical k-nearest neighbours classifier [29]. In WEKA it can select an appropriate value of k based on cross-validation and can also perform distance weighting.

Random Tree is a base classifier in WEKA generating a decision tree by allocating a specified and fixed number of randomly chosen features to each of its node [29]. It performs backfitting by taking into account an estimate of class probabilities based on the training set.

Ridor is a base classifier in WEKA using induction of Ripple Down Rules [29]. First, it generates a default rule, and then the exceptions for the default rule with the least weighted error rate. The exceptions are a set of rules that predict classes other than the default. After that it generates the best exceptions for each exception and iterates until the desired performance is achieved. This process produces a tree-like expansion of exceptions.

Let us refer to [28], [29], [30], [37] for more information on these base classifiers.

5 Standard Ensembles for CBMLDE Classifiers

This section presents succinct background information on the standard ensembles used as building blocks for constructing CBMLDE classifiers in our experiments. These ensembles were chosen, because they are all well known and available in WEKA [28].

AdaBoost is a WEKA implementation of Boosting [29]. It trains several classifiers in succession. Every next classifier is trained on the instances that have turned out more difficult for the preceding classifier. To this end all instances are assigned weights, and if an instance turns out difficult to classify, then its weight is increased at the next boosting step.

Bagging is a standard ensemble classifier in WEKA generating a collection of new sets by resampling the given training set at random and with replacement [29]. These sets are called *bootstrap samples*. New classifiers are then trained, one for each of these new training sets. They are amalgamated via a majority vote.

Decorate is another efficient ensemble classifier available in WEKA [29]. It constructs special artificial training examples to build diverse base classifiers.

Weighted Vote combines the predictions made by several classifiers with assigned weights. Here we used the diagnostic accuracy of each classifier as its weight, so that more accurate classifiers contributed more to the prediction. The weighted vote combines several classifiers to increase their diversity and in doing so improves the accuracy of the overall prediction.

Complete explanations of these ensembles and further references are given in [29], [30], [37], [38].

6 Experiments and Discussion

In this paper we use the diagnostic accuracy for assessing the performance of classifiers, since it is the main measure applied by the medical practitioners in practical work. Tenfold cross validation was employed to prevent overfitting. This is a standard and very well known procedure explained, for example, in [29]. However, in our experiments is had to be applied in the settings of the cloud paradigm with several data centers. Therefore we had to prepare data files arranging them in a special way to fit this paradigm. Here we include explanations of how data were prepared to implement the tenfold cross validation in the different settings of the cloud framework and at the same time to simulate the operation of classifiers in the cloud.

To prepare the dataset for experiments, all instances of data were collected in one csv file. The last column of the file was the class value indicated in the DiScRi database for each instance. Our experiments presented in this section used a binary classification of CAN with two class values 'CAN' and 'no CAN'. Preprocessing was used to reduce of the number of incomplete fields. More than 50 expert editing rules for preprocessing were collected. To automate the application of these expert rules a Python script was written by the third author. Python is a convenient programming language that has been used in many important applications in biomedical informatics, see for example [39]. The majority of the expert editing rules utilized the fact that various medical parameters change only gradually with time, and so their values usually behave as those of an either increasing or decreasing mathematical function. Hence it is safe for data mining purposes to assume that an incomplete field is approximately equal to the average of the preceding and following values of the same field. For other attributes, it is known that some clinical values indicating pathology very seldom improve. For example, if a person has been diagnosed with diabetes, then this diagnosis can be recorded in all subsequent instances of data for the same patient. Finally, some of the expert editing rules checked data for consistency and deduced the values of incomplete fields from other closely related fields. For example, the 'Diagnostic DM (years)' feature in DiScRi refers to the number of years since the patient has been diagnosed with diabetes. If this number is greater than zero in an instance, then the value of another related feature, the 'Diabetic Status', must be set as 'yes'. These editing rules were collected in consultation with the experts managing the database and were included in the Python script for preprocessing data.

Then we divided the file into four csv files denoted by F_C, F_{GP}, F_P, F_E. The last column of all of these files was the same class value column with entries labelled as 'CAN' or 'no CAN' to be used for class prediction. The other columns in these files

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2016.2567389, IEEE Transactions on Cloud Computing

were the chosen cardiology features in F_C, the GP features in F_{GP}, the podiatry features in F_P, and the endocrinology features in F_E. These files were used in all experiments presented in this paper. Next we discuss how each of the experiments was organized to simulate the operation of classifiers in the cloud and apply tenfold cross validation for each experiment.

6.1 Experiment 1

The first experiment investigated all combinations of possible options in CBMLDE classifiers for the cloud model where the data were allocated to 3 data centers in order to choose the best options. We considered all combinations of base classifiers presented in Section 4 deployed at the bottom layer of CBMLDE and ensemble methods presented in Section 5 deployed at the top and middle layer of CBMLDE as explained in Section 2.

To show how to design and train the CBMLDE classifiers in situations where different specialists use CSPs with different data centers, in this experiment we assume that the file F_C is allocated to a separate Cardiology Data Center, and file F_E is allocated to the Endocrinology Data Center. At the same time we also show that the CBMLDE classifier can take into account the possibility of allocating the data from different doctors to one and the same data center. To this end we assume that the GP and Podiatry data F_{GP} and F_P are stored in the same GP and Podiatry Data Center, and so we can combine the files F_{GP} and F_P into one file $F_G = F_{GP} \cup F_P$ containing all GP features and Podiatry features, and only one last column with class values. The distribution of data among the data centers for the first experiment is shown in Figure 1.

Let us now fix a combination of parameters for the CBMLDE classifier C and discuss how the experiment was organized and how the data were prepared to simulate the operation of C in the cloud and carry out tenfold cross validation to prevent overfitting.

The application of tenfold cross validation means that we divided all patients into ten stratified folds and denoted these groups of patients by $P_1, P_2, \ldots, P_{10}$. This division of patients divided each of the data files F_C, F_G, F_E into ten stratified folds and produced 30 files. The file F_C was divided into ten folds $F_{C1}, F_{C2}, \ldots, F_{C10}$, where the file F_{C1} contained all data of the patients of the group P_1 contained in the file F_C, the file F_{C2} contained all data of the patients of the group P_2 contained in the file F_C, and so on. Likewise, the file F_G was divided into ten folds $F_{G1}, F_{G2}, \ldots, F_{G10}$, where the file F_{G1} contained all data of the patients of the group P_1 contained in the file F_G, the file F_{G2} contained all data of the patients of the group P_2 contained in the file F_G, and so on. Similarly, the file F_E was divided into ten folds $F_{E1}, F_{E2}, \ldots, F_{E10}$ too. We used these folds to conduct

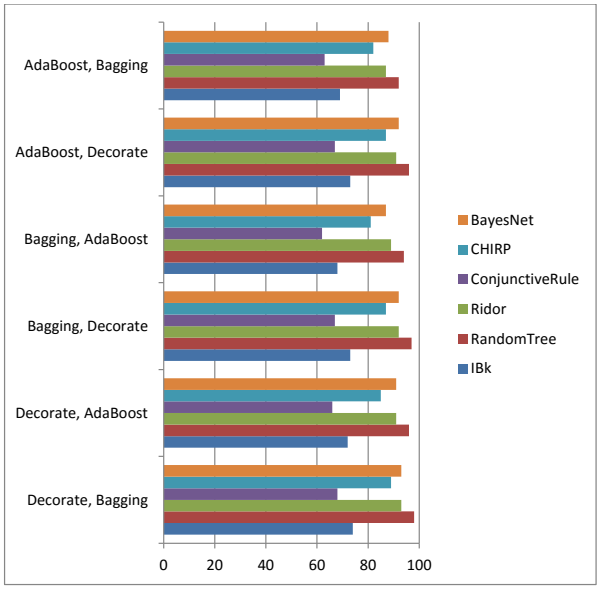

Fig. 3. Experiment 1 comparing the performance of all combinations of possible options for the CBMLDE classifier with 3 data centers.

ten consecutive rounds of tests for each collection of options to be included in the CBMLDE classifier C. Here we discuss how the files were used to apply the tenfold cross validation and determine the average accuracy of C for the diagnosis of CAN in this cloud model.

In the first round of the tenfold cross validation, we used the first group of patients P_1 to divide all files into folds as follows. New data file

$$T_{C1} = F_{C2} \cup F_{C3} \cup \ldots \cup F_{C10} \quad (3)$$

was used as the training file to train all classifiers allocated to the Cardiology Data Center and generate the models of these classifiers. This means that the training set T_{C1} contains all records of the file F_C that do not belong to the first fold F_{C1}. Likewise, the data files

$$T_{G1} = F_{G2} \cup F_{G3} \cup \ldots \cup F_{G10}, \quad (4)$$

$$T_{E1} = F_{E2} \cup F_{E3} \cup \ldots \cup F_{E10} \quad (5)$$

were used to train all classifiers allocated to the GP and Podiatry Data Center and to the Endocrinology Data Center, respectively. This means that the training set T_{G1} contains all records of the file F_F that do not belong to F_{G1}, and the training set T_{E1} comprises all records of the file F_E that do not belong to F_{E1}.

All the building blocks of C were trained on the training files T_{C1}, T_{G1}, T_{E1} allocated to the same node of the cloud. The accuracies of the diagnosis achieved by these parts on the training sets were used as the weights for the WV. These weights were communicated to the WV for combining the outputs

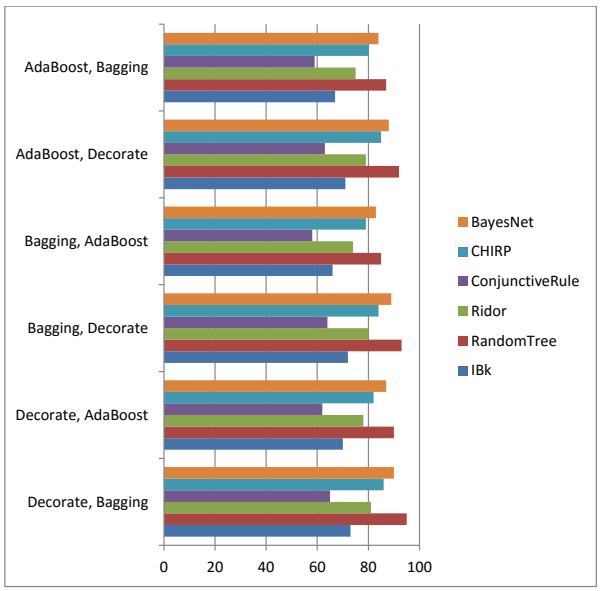

Fig. 4. Experiment 2 comparing the performance of all combinations of possible options for the CBMLDE classifier with 6 data centers.

of the parts of C allocated to the three nodes of the cloud. This completed the training stage.

For testing the work of C in the first round of tenfold cross validation, three validate files V_C, V_G and V_E were created at the three nodes of the cloud. Each of them was equal to the corresponding fold of the set at the same node of the cloud, so that $V_C = T_{C1}$, $V_G = TG1$ and $V_E = T_{E1}$. Each instance of data for every patient was now divided and recorded in these three files V_C, V_G and V_E. These validate files were used for testing the trained model of C. Each part of C was applied to the validate file that belonged to the same node of the cloud, and the outputs were combined by the weighted vote. The predicted CAN class was then compared with the correct diagnosis, and the accuracy was recorded as the outcome of C at the first round of the tenfold cross validation.

The other nine consecutive rounds of the tenfold cross validation proceeded for B and C in the same way using the groups of patients $P_2, \ldots, P_{10}$ in each round, respectively, in the same way as the first group P_1 was used in the first round. The average accuracy achieved in all ten consecutive rounds was the final outcome of tenfold cross validation for C.

All figures presenting the results of our experiments contain the diagnostic accuracy as the main measure of performance. The diagnostic *accuracy* is defined as the percentage of all patients diagnosed correctly. It can be expressed as the probability that the prediction of the classifier for an individual patient is correct. We used WEKA to train and test classifiers and ensemble classifiers.

We carried out this testing for all combinations of the options available in setting up the CBMLDE classifier – all base classifiers presented in Section 4 and standard ensembles listed in Section 5 deployed in CBMLDE as explained in Section 2. The results of these tests are presented in Figure 3. This figure contains the average performance against the validate sets generated for the stratified tenfold cross validation as discussed above. Figure 3 demonstrates that the best result was obtained by Decorate in the top layer, Bagging at the middle layer, and Random Tree as the base classifier.

6.2 Experiment 2

The second experiment investigated all combinations of options for setting up CBMLDE classifier in the cloud model where the data were allocated to 6 data centers illustrated in Figure 2. Again, our experiment explored all combinations of base classifiers presented in Section 4 deployed at the bottom layer of CBMLDE and ensemble methods presented in Section 5 deployed at the top and middle layer of CBMLDE as explained in Section 2.

To prepare data for the setting illustrated in Figure 2, we divided all patients into two approximately equal parts P_1 and P_2. Patients of P_1 were allocated to Location 1 of Figure 2, and their data were used there in a way analogous to the use of all data was utilized in the first experiment. Patients of P_2 were allocated to Location 2 of Figure 2, and their data were used there in a similar way. This means that both sets P_1 and P_2 were divided into ten stratified folds so that

$$P_1 = P_{1,1} \cup P_{1,2} \cup \ldots \cup P_{1,10}, \quad (6)$$

$$P_2 = P_{2,1} \cup P_{2,2} \cup \ldots \cup P_{2,10}. \quad (7)$$

The union $P_{1,1} \cup P_{2,1}$ of the first folds of patients at both locations was used to define the training set and the validate set for the first round of tenfold cross validation, and so on. The average accuracies achieved during ten rounds are presented in Figure 4 for all combinations of options in setting up the CBMLDE classifiers. Figure 4 demonstrates that the best result was obtained by CBMLDE classifier with Decorate at the top layer, Bagging at the middle layer, and Random Tree as the base classifier.

6.3 Experiment 3

The third experiment investigated all combinations of options for setting up CBMLDE classifier in the cloud model where data were allocated to 9 data centers shown in Figure 5. Here we divided patients into 3 approximately equal groups, then divided each of the three groups into ten stratified folds and used them to prepare all data files as above. The results of experiment with 9 data centers are given in Figure 6, which demonstrates that the best outcome was again obtained by CBMLDE classifier with Decorate at the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2016.2567389, IEEE Transactions on Cloud Computing

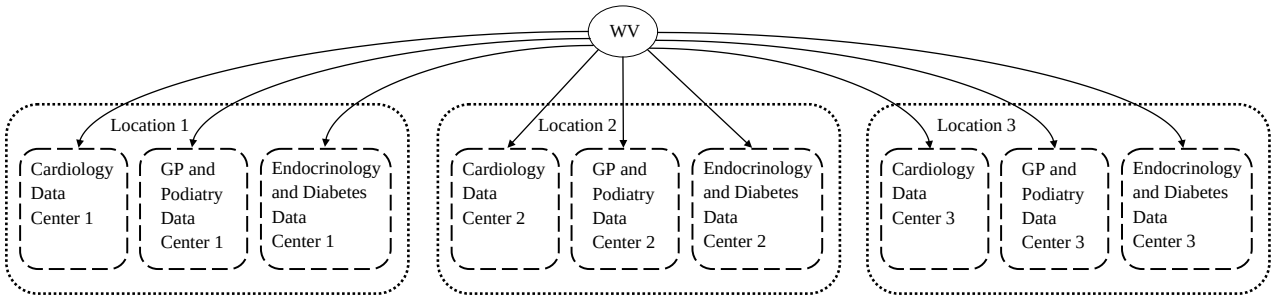

Fig. 5. CBMLDE classifier with nine data centers.

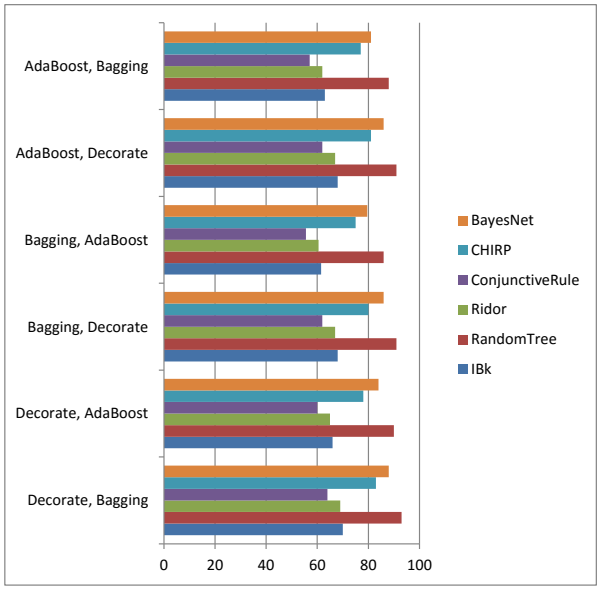

Fig. 6. Experiment 3 comparing all possible combinations of options for the CBMLDE classifier with 9 data centers.

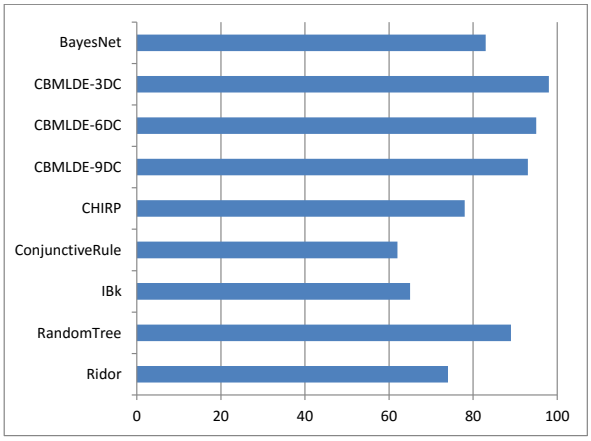

Fig. 7. Experiment 4 comparing the effectiveness of CBMLDE classifier and several base classifiers. The CBMLDE classifier was tested for all three cloud frameworks presented in Figures 1, 2, and 5 with 3, 6 and 9 data centers. The base classifiers operated in all of these frameworks in the same way by collecting all data in one data center and then applying the base classifier.

top layer, Bagging at the middle layer, and Random Tree as the base classifier.

6.4 Experiment 4

The fourth experiment was designed to compare the performance of CBMLDE with several well known classifiers representing the most important and well known types of machine learning algorithms. We compared CBMLDE classifier with the following base classifiers presented in Section 4: BayesNet, CHIRP, ConjunctiveRule, IBk, Random Tree, Ridor. Three cloud frameworks presented in Figures 1, 2, and 5 with 3, 6 and 9 data centers were tested. The best options of CBMLDE classifier for these frameworks with 3, 6 and 9 data centers given in are denoted by CBMLDE-3DC, CBMLDE-6DC and CBMLDE-9DC, respectively.

On the other hand, in Experiment 4 each base classifier performed in the same way in all of these three frameworks, as follows. It was allocated to only one node of the cloud and it had to transfer all data to the same node and process all data there. This is why all base classifiers obtained the same results in all three framework of Experiment 4. Their results are shown in Figure 7. These best outcomes obtained by CBMLDE-3DC, CBMLDE-6DC and CBMLDE-9DC are copied from Figures 3, 4 and 6 into Figure 7 for ease of comparison. Figure 7 demonstrates that CBMLE classifier not only has completely eliminated the need in data transfer to one node of the cloud, but also has significantly outperformed all base classifiers in all three frameworks.

6.5 Experiment 5

The last experiment was designed to compare the performance of CBMLDE with a simpler counterparts using base classifiers combined with a simple majority vote. We explored all base classifiers presented in

Section 4: BayesNet, CHIRP, ConjunctiveRule, IBk, Random Tree, Ridor. Each of these base classifiers in turn was used to set up a simple counterpart model applying weighted vote (WV). The corresponding models are denoted by WV+BayesNet, WV+CHIRP, WV+ConjunctiveRule, WV+IBk, WV+Random Tree, WV+Ridor in order to present their results.

To set up a simple model WV+B using base classifier B, a copy of B was allocated to each of the nodes of the cloud. These copies of B were trained on the training files allocated to the same node of the cloud. The accuracies of the diagnostics achieved by these copies on the training sets were then used as the weights for the WV. These weights were communicated to the WV for combining the outputs of the three base classifiers from different nodes. The validate set was then processed using the same weights that we determined by the training set.

We compared the best versions of CBMLDE classifier with outcomes obtained by all models WV+BayesNet, WV+CHIRP, WV+ConjunctiveRule, WV+IBk, WV+Random Tree, WV+Ridor. These results are presented in Figure 8, which demonstrates that CBMLE classifier has significantly outperformed all counterpart models in all three cloud frameworks.

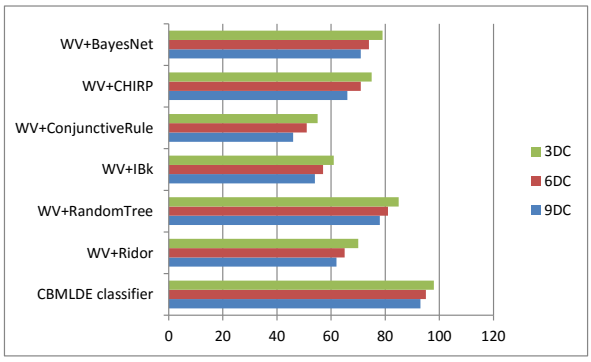

Fig. 8. Experiment 5 comparing the effectiveness of CBMLDE classifier and several simpler counterpart classifiers applying the weighted vote directly to the corresponding base classifiers allocated to data centers. Three cloud frameworks presented in Figures 1, 2, and 5 with 3, 6 and 9 data centers were tested.

7 Conclusion

The model of CBMLDE classifiers eliminates the need to transfer large amounts of patient data in the data center network for training of the classifiers. Our experiments simulated the use of CBMLDE classifiers in the cloud and determined that the best outcomes are obtained by the CBMLDE classifier combining Weighted Vote, Decorate, Bagging and Random Tree. The results of our comprehensive collection of tests show that the best models of CBMLDE not only completely eliminate the need in patient data transfer, but also have significantly outperformed all base classifiers and counterpart models in all three cloud frameworks.

Acknowledgments

The authors are grateful to three reviewers for comments and corrections that have helped to improve this article.

References

[1] R. Buyya, C. Vecchiola, and T. Selvi, *Mastering Cloud Computing*. Burlington, Massachusetts, USA: Morgan Kaufmann, 2013.

[2] P. K. Tysowski and M. A. Hasan, "Hybrid attribute- and re-encryption-based key management for secure and scalable mobile applications in clouds," *IEEE Transactions on Cloud Computing*, vol. 1, pp. 172–186, 2013.

[3] K. Bilal, M. Manzano, S. U. Khan, E. Calle, K. Li, and A. Y. Zomaya, "On the characterization of the structural robustness of data center networks," *IEEE Transactions on Cloud Computing*, vol. 1, pp. 64–77, 2013.

[4] M. H. Ferdaus, M. Murshed, R. N. Calheiros, and R. Buyya, "Virtual machine consolidation in cloud data centers using ACO metaheuristic," in *20th International Conference on Parallel Processing, Euro-Par 2014*, ser. LNCS, vol. 8632, 2014, pp. 306–317.

[5] M. A. Rodriguez and R. Buyya, "Deadline based resource provisioning and scheduling algorithm for scientific workflows on clouds," *IEEE Transactions on Cloud Computing*, vol. 2, pp. 222–235, 2014.

[6] S. K. Garg, A. N. Toosi, S. K. Gopalaiyengar, and R. Buyya, "SLA-based virtual machine management for heterogeneous workloads in a cloud datacenter," *Journal of Network and Computer Applications*, vol. 45, pp. 108–120, 2014.

[7] F. Martin-Sanchez, G. Lopez-Campos, and K. Gray, *Biomedical Informatics Methods for Personalized Medicine and Participatory Health*. Elsevier, 2013, ch. in "Methods in Biomedical Informatics: A Pragmatic Approach", pp. 347–394.

[8] A. J. Koutsoukis, G. H. Lopez-Campos, and F. Martin-Sanchez, "Merging personalized and participatory medicine: Interpretation of individual genomes," *Studies in Health Technology and Informatics*, vol. 202, pp. 24–27, 2014.

[9] G. Lopez-Campos, R. Bellazzi, and F. Martin-Sanchez, "INDIV-3D. A new model for individual data integration and visualisation using spatial coordinates," *Studies in Health Technology and Informatics*, vol. 190, pp. 172–174, 2013.

[10] J. Kilby, K. Gray, K. Elliott, J. Waycott, F. M. Sanchez, and B. Dave, "Designing a mobile augmented reality tool for the locative visualisation of biomedical knowledge," *Studies in Health Technology and Informatics*, vol. 192, pp. 652–656, 2013.

[11] H. F. Jelinek, J. H. Abawajy, A. V. Kelarev, M. U. Chowdhury, and A. Stranieri, "Decision trees and multi-level ensemble classifiers for neurological diagnostics," *AIMS Medical Science*, vol. 1, pp. 1–12, 2014.

[12] W. Yao, J. He, G. Huang, J. Cao, and Y. Zhang, "Personalized recommendation on multi-layer context graph," *Lecture Notes in Computer Science*, vol. 8180, pp. 135–148, 2013.

[13] J. H. Abawajy, A. V. Kelarev, and M. Chowdhury, "Multistage approach for clustering and classification of ECG data," *Computer Methods and Programs in Biomedicine*, vol. 112, pp. 720–730, 2013.

[14] J. He, S. Du, Z. Wang, Z. Wang, J. Zhou, and Q. Lou, "Linearly-polarized short-pulse AOM Q-switched 978 nm photonic crystal fiber laser," *Optics Express*, vol. 21, pp. 29 240–29 245, 2013.

[15] W. Yao, J. He, G. Huang, J. Cao, and Y. Zhang, "A graph-based model for context-aware recommendation using implicit feedback data," *World Wide Web*, vol. 18, pp. 1351–1371, 2015.

[16] A. Stranieri, J. Abawajy, A. Kelarev, S. Huda, M. Chowdhury, and H. F. Jelinek, "An approach for Ewing test selection to support the clinical assessment of cardiac autonomic neuropathy," *Artificial Intelligence in Medicine*, vol. 58, pp. 185–193, 2013.

[17] L. Wang, J. C. Bezdek, C. Leckie, and R. Kotagiri, "Selective sampling for approximate clustering of very large data sets," *International Journal of Intelligent Systems*, vol. 23, pp. 313–331, 2008.

[18] L. Wang, C. Leckie, R. Kotagiri, and J. Bezdek, "Approximate pairwise clustering for large data sets via sampling plus extension," *Pattern Recognition*, vol. 44, pp. 222–235, 2011.

[19] F. Li, J. He, G. Huang, Y. Zhang, and Y. Shi, "A clustering-based link prediction method in social networks," *Procedia Computer Science*, vol. 29, pp. 432–442, 2014.

[20] P. B. Perez Villamil, A. Romera Lopez, S. Hernandez Prieto, G. Lopez Campos, A. Calles, J. A. Lopez Asenjo, J. Sanz Ortega, C. Fernandez Perez, J. Sastre, R. Alfonso, T. Caldes, F. Martin Sanchez, and E. Diaz Rubio, "Colon cancer molecular subtypes identified by expression profiling and associated to stroma, mucinous type and different clinical behavior," *BMC Cancer*, vol. 12, pp. 1–16, article number 260, 2012.

[21] X. Yin, B.-H. Ng, J. He, Y. Zhang, and D. Abbott, "Accurate image analysis of the retina using hessian matrix and binarisation of thresholded entropy with application of texture mapping," *PLoS ONE*, vol. 9, pp. 1–17, article number e95 943, 2014.

[22] J. Chan, N. X. Vinh, W. Liu, J. Bailey, C. A. Leckie, R. Kotagiri, and J. Pei, "Structure-aware distance measures for comparing clusterings in graphs," *Lecture Notes in Artificial Intelligence*, vol. 8443, pp. 362–373, 2014.

[23] G. Huang, J. He, Y. Zhang, W. Zhou, H. Liu, P. Zhang, Z. Ding, Y. You, and J. Cao, "Mining streams of short text for analysis of world-wide event evolutions," *World Wide Web*, vol. 18, pp. 1201–1217, 2015.

[24] R. Hassan, M. Hossain, J. Bailey, G. Macintyre, J. W. K. Ho, and R. Kotagiri, "A voting approach to identify a small number of highly predictive genes using multiple classifiers," *BMC Bioinformatics*, vol. 10, pp. 1–12, article number S19, 2009.

[25] R. Islam, W. Zhou, and M. U. Chowdhury, "Email categorization using (2+1)-tier classification algorithms," in *Proceedings – 7th IEEE/ACIS International Conference on Computer and Information Science, IEEE/ACIS ICIS 2008, In conjunction with 2nd IEEE/ACIS Int. Workshop on e-Activity, IEEE/ACIS IWEA 2008*, 2008, pp. 276–281.

[26] K. K. Gupta, B. Nath, and R. Kotagiri, "Layered approach using conditional random fields for intrusion detection," *IEEE Transactions on Dependable and Secure Computing*, vol. 7, pp. 35–49, 2010.

[27] A. Goder and V. Filkov, "Consensus clustering algorithms: comparison and refinement," in *Tenth SIAM Workshop on Algorithm Engineering and Experiments, ALENEX 2008*. San Francisco, January 19, 2008: Society for Industrial and Applied Mathematics, 2008, pp. 109–117.

[28] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *SIGKDD Explorations*, vol. 11, pp. 10–18, 2009.

[29] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed. Amsterdam: Elsevier/Morgan Kaufman, 2011.

[30] R. R. Bouckaert, E. Frank, M. Hall, R. Kirkby, P. Reutemann, A. Seewald, and D. Scuse, "WEKA manual for version 3-7-12," http://www.cs.waikato.ac.nz/ml/weka/, viewed 15 January, 2015.

[31] A. Fahad, N. Alshatri, Z. Tari, A. Alamri, I. Khalil, A. Zomaya, S. Foufou, and A. Bouras, "A survey of clustering algorithms for big data: Taxonomy & empirical analysis," *IEEE Transactions on Emerging Topics in Computing*, vol. online first, pp. to appear soon, DOI:10.1109/TETC.2014.2 330 519, 2014.

[32] D. Cornforth and H. F. Jelinek, "Automated classification reveals morphological factors associated with dementia," *Applied Soft Computing*, vol. 8, pp. 182–190, 2007.

[33] D. J. Ewing, J. W. Campbell, and B. F. Clarke, "The natural history of diabetic autonomic neuropathy," *Q. J. Med.*, vol. 49, pp. 95–100, 1980.

[34] D. J. Ewing, C. N. Martyn, R. J. Young, and B. F. Clarke, "The value of cardiovascular autonomic function tests: 10 years experience in diabetes," *Diabetes Care*, vol. 8, pp. 491–498, 1985.

[35] A. H. Khandoker, H. F. Jelinek, and M. Palaniswami, "Identifying diabetic patients with cardiac autonomic neuropathy by heart rate complexity analysis," *BioMedical Engineering OnLine*, vol. 8, pp. 1–12, 2009.

[36] D. J. Ewing and B. F. Clarke, "Diagnosis and management of diabetic autonomic neuropathy," *British Medical Journal*, vol. 285, pp. 916–918, 1982.

[37] M. Negnevitsky, *Artificial Intelligence: A Guide to Intelligent Systems*, 3rd ed. New York: Addison Wesley, 2011.

[38] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. New York: Wiley Interscience, 2004.

[39] H. K. Ho, M. J. Kuiper, and R. Kotagiri, "PConPy – a Python module for generating 2D protein maps," *Bioinformatics*, vol. 24, pp. 2934–2935, 2008.

Morshed U. Chowdhury received his PhD from Monash University, Australia in 1999. Dr. Chowdhury is an academic staff member in the School of Information Technology, Deakin University, Australia. Prior to joining Deakin University, he was an academic staff in Gippsland School of Computing and Information Technology, Monash University, Australia. Dr. Chowdhury has more than 12 years of industry experience in Bangladesh and Australia. As an International Atomic Energy Agency (IAEA) fellow he has visited a number of International Laboratory/Centers such as Bhaba Atomic Research Centre, India, and Brookhaven National Laboratory, New York, USA, International Centre for Theoretical Physics (ICTP)-Italy. Dr. Chowdhury's current research interests are RFID security, wireless network security and security of social networks, documentation security etc. He has published more than hundred five research papers including a number of journal papers, conference papers and book chapters. He has organized a number of international conferences and served as a member of the technical and program committee of several international conferences since 2001. He has also acted as reviewer of many journal papers.

Jemal H. Abawajy is currently a full Professor and the Director of the Parallel and Distributing Computing Lab, Deakin University, Burwood, Australia. He was a member of the organizing committees for more than 300 international conferences serving in various capacities including chair and general cochair. He has published more than 300 refereed articles, supervised numerous Ph.D. students to completion and is on the editorial boards of many journals.

Andrei Kelarev is an author of two books and 197 journal articles. He worked as an Associate Professor in the University of Wisconsin and University of Nebraska in USA and as a Senior Lecturer in the University of Tasmania in Australia. He was a Chief Investigator of a large Discovery grant from Australian Research Council, an editor of five journals, and a member of the program committees of several conferences. He is now with the Parallel and Distributed Computing Lab at Deakin University, Australia.

Herbert Jelinek received the B.Sc. (Hons.) degree in human genetics from the University of New South Wales, Sydney, Australia, followed by a Graduate Diploma in neuroscience from the Australian National University, Canberra, Australia, and his PhD. degree in medicine from the University of Sydney, Australia. He is a honorary Clinical Associate Professor with the Australian School of Advanced Medicine, Macquarie University, Sydney, Australia, and an Associate Professor of the Centre for Research in Complex Systems, Charles Sturt University, Albury, Australia. Dr Jelinek has been organizing a rural diabetes complications screening research project for over ten years in Australia and has published widely in ECG signal processing and diabetic retinopathy image analysis as well as data mining applications of biomarkers associated with diabetes disease progression. His current research interests include neurogenetics of diabetes and cognitive function. He is a member of the IEEE Biomedical Engineering Society and the Australian Diabetes Association.