# Latency-based Analytic Approach to Forecast Cloud Workload Trend for Sustainable Datacentres

Yao Lu, *Member, IEEE,* Lu Liu, *Member, IEEE,*
John Panneerselvam, *Member, IEEE,* Xiaojun Zhai, *Member, IEEE,* Xiang Sun, Nick Antonopoulos, *Member, IEEE*

**Abstract**—Cloud datacentres are turning out to be massive energy consumers and environment polluters, which necessitate the need for promoting sustainable computing approaches for achieving environment-friendly datacentre execution. Direct causes of excess energy consumption of the datacentre include running servers at low level of workloads and over-provisioning of server resources to the arriving workloads during execution. To this end, predicting the future workload demands and their respective behaviours at the datacentres are being the focus of recent research in the context of sustainable datacentres. But prediction analytics of Cloud ds suffer various limitations imposed by the dynamic and unclear characteristics of Cloud workloads. This paper proposes a novel forecasting model named K-RVLBPNN (K-means based Rand Variable Learning Rate Backpropagation Neural Network) for predicting the future workload arrival trend, by exploiting the latency sensitivity characteristics of Cloud workloads, based on a combination of improved K-means clustering algorithm and BPNN (Backpropagation Neural Network) algorithm. Experiments conducted on real-world Cloud datasets exhibit that the proposed model exhibits better prediction accuracy, outperforming traditional Hidden Markov Model, Naïve Bayes Classifier and our earlier RVLBPNN model respectively.

**Index Terms**—Clustering, Energy-aware systems, Data Models, Pattern Analysis

## 1 Introduction

The evolution of Cloud Computing technology in the recent years has led to the development of various effective means of service provisioning, such that services proving to be previously costly and difficult to reach has now made easily available to all clients. Due to the easy access, low-cost, pay-as-you-go models of Cloud Computing services, the number of users using Cloud Computing both individuals and from small-scale to large-scale industries has increased unprecedentedly in the past few years [1]. The on-demand resource provisioning nature of Cloud datacentres help customers to obtain the resources including CPU, memory, and network at a reasonable cost, in order to carry out their job executions without any interruptions [2]. On the contrary, the increasing number of Cloud datacentres are addressed to be causing increased environmental implications [3] as datacentres release excessive amounts of carbon footprints. Furthermore, Cloud data-centres consume enormous amounts of electricity to run their servers for hosting workload execution, and for cooling management. Although the concept of sustainable computing is emerging, which in essence utilises renewable energy sources as input power, such a kind of environment-friendly datacentres are still significant in existence [4].

Owing to such environment degrading characteristics of datacentres, reducing the energy and environment implications of Cloud datacentres through various strategic approaches [5] [6] has been one of the primary focuses of recent research in the context of Cloud Computing. It is worthy of note that the cooling systems of a typical Cloud datacentre can consume a significant proportion of the energy actually those utilised during the actual datacentre execution. Such additional costs of running a datacentre might also affect the economic benefits of the service providers [7].

The current level of workloads executed at the servers, and the level of CPU and memory consumption of the workloads usually have a direct impact on the level of energy consumed by the server resources. In fact, processors [8] are the addressed to be the largest energy consumers in a server. Workload can be of various types in terms of their resource intensiveness such as CPU-intensive, memory-intensive, both CPU and memory-intensive and network intensive. Workloads requiring larger CPU resources are usually energy-intensive in comparison to the memory-intensive workloads. One of the reasons for the excess energy consumption of the datacentres is over-provisioning of resources to process workloads at the servers. In this way, service providers usually provision resources to workloads at a level that far exceed their actual requirements. Strategic approaches widely adopted by the existing state-of-the-art energy saving techniques include predicting the actual de-

- *Yao Lu, Lu Liu, John Panneerselvam, Xiang Sun and Nick Antonopoulos are with the Department of Electronics, Computing and Mathematics, College of Engineering and Technology, University of Derby, Kedleston Road, Derby, United Kingdom, DE22 1GB; Xiaojun Zhai is with the School of Computer Science and Electronic Engineering at the University of Essex, Colchester UK; Yao Lu and Xiang Sun are also with School of Computer Science and Telecommunication Engineering, Jiangsu University, Jiangsu, China.*

- *E-mail: {y.lu2, l.liu, j.Panneerselvam, n.Antonopoulos}@derby.ac.uk, xzhai@essex.ac.uk, x.sun@ derby.ac.uk.*
- *Lu Liu is the corresponding author*

mands of the workloads with the motivation of provisioning resources to workloads at an appropriate level that can characterise low energy wastage profile. But this prediction is a complex process, since the cloud workloads characterise an increased level of dynamism [9] in their actual behaviours at the datacentres in terms of their resource consumption level. To this end, resource scheduling and resource provisioning mechanisms in Cloud datacentres naturally involve various complications.

The running duration of the Cloud workloads can also have an impact on its energy demands. Thus predicting the runtime duration of the arrived workloads might provide additional insights to infer their energy consuming nature. In most cases, Cloud workloads characterise a shorter duration and arrive more frequently. In fact, the arrival of the Cloud workloads can also be related to the operating business hours. Duration of the workloads usually determine their latency-sensitivity [10], which is usually a parameter that defines the time-scale within which the workload must be processed by the providers after its arrival. Workloads with an increased level of latency sensitivity usually requires a less processing time and vice versa. Despite the existing works on Cloud research, further research and analysis of Cloud entities is still crucial. This paper is aimed at building a scientific model based on user behaviours in terms of their current job submission pattern and their corresponding workload behaviours, along with their historical workload behaviours, in order to predict their future behaviours anticipated at the datacentres. A reliable predictive model should accurately reflect the main characteristics of datacentres, users and workloads to achieve accurate prediction results [11]. Such characteristics can be summarised as follows: Firstly, increasing number of arriving workloads demand increased number of server resources. But, turning on more number of server resources during less number of arriving workloads might result in wastage of server resources, by the way of feeding such servers with electricity but without extracting any information services. Secondly, Cloud resources can behave dynamically, in such a way that similar workloads might consume varied level of resources at the datacentre during their actual execution. This implies that Cloud workload behaviours cannot be generalised, and should be treated uniquely. Thirdly, Cloud workload arrival trend can take dynamic shift in time. Although, Cloud workload characterise a certain level of periodicity in relation to the operating business hours, the arrival trend can change drastically under a short time interval.

Despite the existing works on characterising Cloud workloads [9], [12], [13], the dynamism and nature of the Cloud workloads are still not clear and explicit. Such, speculations are important since the prediction models are developed based on them. Inaccurate characterisation of the Cloud workloads might lead to wrong prediction, which might further mislead the level of resources provisioned to process the workloads. There are two immediate implications. While an over-provisioned level of resources cause energy wastage, under-provisioned resources might lead to the termination of workloads due to resource scarcity. Increased number of job terminations could violate the Service Level Agreement (SLA), which is usually initially agreed between the clients and the providers, and directly affect the Quality of Service (QoS). To this end, with the motivation of reducing the prediction inaccuracies, this paper proposes a novel forecasting model named K-RVLBPNN, based on an improved K-means clustering algorithm and BP Neural Network, in order to predict the anticipated level of future-service requests with reliable level of accuracy. The proposed model exploits the latency sensitivity levels of the workloads to predict the service request frequency anticipated in the future. Important contributions of this paper are listed as follows:

- Firstly, we improve the traditional K-means algorithm by optimising its clustering performance through capturing the latency sensitivity of the Cloud workloads. This enhances the suitability of the traditional K-means algorithm to handle large-scale dynamic Cloud workloads.
- Secondly, we develop a novel workload prediction model called K-RVLBPNN. The core of K-RVLBPNN model is the combination of our improved K-means algorithm and random Back Propagation (BP) neural network. This model can efficiently predict the future workload trend by exploiting historical data clustered by the improved K-means algorithm.
- Thirdly, we implement our proposed K-RVLBPNN model on a real-world Cloud dataset and performed extensive evaluations against the existing Hidden Markov Model, Naïve Bayes Classifier and RVLBPNN model respectively. Experimental results demonstrate that our proposed forecasting model can achieve higher prediction accuracy whilst estimating the future service requests in Cloud datacentre to aid sustainable datacentres.

The rest of this paper is organised as follows: Section 2 presents the related works of Cloud workload forecasting models. Section 3 presents a background study on Cloud workload characteristics and their latency sensitivity. Section 4 proposes our novel prediction model based on the improved K-means clustering algorithm and BP Neural Network. Our experimental evaluations are presented in Section 5 and Section 6 concludes this paper along with outlining our future research directions.

## 2 RELATED WORK

A wide range of research works have been proposed to accelerate the rational use of cloud computing resources, and can be predominantly categorised into hardware-based approaches and techniques at the software using strategic approaches. DVFS (Dynamic Voltage and Frequency Scaling) and DPM (Dynamic Power Management) have been the focus of the work proposed in [14] to reduce the wastage of the server's resources. DVFS based algorithms rely on the server's performance for adjusting the processor supply voltage and frequency to reduce the overall power consumption, in accordance with the arrival trend of the workloads and the resource demands. DMP algorithm can switch servers running on low workload levels for a relatively longer time into energy saving modes to save energy, whilst finishing all tasks within the deadlines. However, DVFS method has only been suitable to reduce the dynamic server power by altering the frequency and voltage, which is usually proportional to the server utilisation level, and ignores the leakage currents of the servers. On the other hand, DMP based techniques can effectively reduce the static leakage currents, and ignores the dynamic power. It is important for an energy saving approach to act upon both the static and dynamic power simultaneously, insights on the trend of workload arrival trend can aid to achieve this objective. It is quite challenging for an approach to coordinate both static and dynamic power for achieving efficient energy conservation.

Workload prediction techniques have been aiding Cloud datacentres for workload management, resources allocation, optimising servers and so on. Predicting the future arrival scale of workloads can help managing the server farm well by the way of turning ON/OFF the required number of servers. A hybrid workload forecasting method called NUP [15] has been proposed to provide inferences on the type of arriving workloads based on Auto-correlation Coefficient and Hurst Exponents. Once the type of workloads has been acknowledged, two different forecasting algorithms have been applied to deal with the corresponding Cloud workloads respectively. In the prediction strategy of NUP, linear regression and ARMA (Auto Regressive Moving Average) model have been integrated to predict the arrival trend of Cloud workloads, and further SVM (Support Vector Machine) has been used for characterising the periodicity among the arriving Cloud workloads. However, NUP cannot be utilised to predict other types of Cloud workloads characterising less or no level of periodicity such as paroxysmal workloads. The works of [16] proposed a method to predict the future workflow based on a fragment database. In this method, historical traces of workloads have been chosen as background workloads to provide inferences on the future trend. These workloads are stored in the database in the form of fragments, and currently arriving trend of workloads have been compared with the historical trend utilising the fragment database in order to evaluate their similarities. If the current trend exhibits higher level of similarity with the historical trend, then the corresponding historical trace has been used as a reference sample for estimating the future workload trend.

The works of [17] developed a prototype model using machine learning techniques to predict the incoming workload trend. The predicted trend has then been compared and moderated with the historical trend to improve the prediction accuracy. This approach helps to visualize the future resource requirements on VMs of users and to allocate resources based on the predicted user's demands. This model identifies suitable scenario from the past to evaluate the appropriateness of the allocated level of resources at a given time. Idle VMs or VMs with lower level of workloads have then been recommended to shut down to save energy and resources according to the predicted state of VMs. Major attributes in this method have been chosen by Pearson Relation method. A lot of traditional machine learning methods have been proposed to predict the future states of VMs. However, most of them incorporate time-consuming algorithms which adds outrageous time overheads. Besides, prediction accuracy has been the only focus of such methods. For instance, prediction technique with good accuracy but incurring significant time-cost might not present an optimum solution for ontime server management.

An Exponential Smoothing (ES) [18] based method exploiting historical insights has been proposed to forecast the future arrival trend of workloads. The proposed mechanism incorporates concurrent iterations to provide a reliable level of prediction accuracy, but this method requires larger historical traces, thus costing more storage space. The works of [19] proposed an AR (Auto Regression) based workload prediction model using a cyclic computation of the arrival trend. Though AR [20] technique has been adopted widely, this methodology has an insurmountable shortcoming. In the process of recursion, prediction errors will also be accumulated. This is to say that the error margin will increase with an extended prediction time. Another significant drawback of the AR prediction model is that it might become less efficient for workloads with less level of inherent periodicity.

The works of [21] proposed a workload forecasting model called CloudInsight, exploiting the combined ability of multiple workload predictors. This prediction frame uses a multi-class regression, where the weights of every predictor determine the prediction accuracy of current workload trends.Though this method enhances the forecasting accuracy of real-time workloads, it incurs significant time overheads whilst choosing a suitable predictor.

The works of [22] developed a prediction framework based on GA (Genetic Algorithm) to forecast the resource requirements of workloads arriving in the next time slot, according to the historical traces of the previous time slot. In their simulation-based analysis, their proposed GA model proves to be a better solution for workload resource prediction under both stable and unstable utilisation tendency. If the previous workload sample do not characterise a similar trend to the workloads in the current time slot, this model can suffer significant prediction inaccuracies.

Recently, a workload prediction model named RVLBPNN [23] has been proposed in our earlier work. This model can forecast the future workload trend by exploiting historical data based on Neural Networks. This model can effectively capture the similarity between successive workflow without the need for marking the characteristics of workloads. In addition, the use of random learning rate allows the model to avoid local minima as much as possible, thus reducing the error of prediction. This model has improved the prediction accuracy of Cloud workloads to some extent in large-scale datacentres. However, this model relies on a manual classification of workloads based on their latency sensitivity, which restrains its deployment in a real-life Cloud environment. In summary, Cloud Computing still demands a smart prediction model that can effectively analyse the characteristics of the Cloud entities to deliver a reliable prediction of the workload arrival trend. With this in mind, this paper proposes a new prediction model named K-RVLBPNN based on an improved K-means algorithm and Neural Networks. Exploiting the workload characteristics and with an automated classification of workloads, our proposed model can potentially deliver a more accurate prediction.

## 3 BACKGROUND

Workloads are usually the user requests arriving at the datacentre for processing in the form of jobs. A single Cloud job can encompass one to several number of tasks [20], such workloads are processed in the VMs hosted in the servers of the datacentres. During execution workloads consume CPU and memory resources of the server resources based on their resource intensiveness. Tasks within a single workload can behave differently at the datacentre in terms of their resource consumption level and duration. Cloud workloads are extremely dynamic in terms of their actual behaviour at the datacentre, such that similar workloads and tasks may behave differently. Based on the behaviours of the tasks, server profile exhibits more fluctuation [24], [25] in terms of their CPU and memory utilisation. Tasks are also bound to have varied service requirements such as throughput, latency, and jitter etc.

Mostly Cloud workloads are usually driven by the operating business hours and thus could characterise notable level of periodicity exhibiting repeating patterns [26] in their arrival frequency. It is worthy of note that the arrival trend could characterise unexpected increase and decline [27] within a shorter time-scale. The relationships [28] between the workloads and users can provide important insights for prediction analytics. Such relationship could remain static [29] for a longer time-scale and can have a significantly positive impact on the prediction accuracy.

Based on the arrival frequency, Cloud workloads have been categorized into five major types [13] as static, periodic, unpredictable, continuously changing, and once-in-a-lifetime workloads. Further, cloud workloads have been characterised as bound to various types of latency sensitivity depending on their requirements of execution time. Such latency sensitivity depends on the Round Trip Time (RTT) [30], [31], which is usually the wait time for the users to receive the execution response. Latency of the workload execution can be dominated by the server characteristics [32] such as CPU and memory capacity, operating system, server workload level and the nature of the workloads etc.

The taxonomy of the latency levels of the Cloud workloads has been extensively studied in the works of [33], [34], such taxonomy have been attributed from level 0 representing the least latency sensitive tasks to level 3 representing the most latency sensitive tasks. Least latency sensitive tasks (level 0) are non-production tasks [33] usually characterise an increased RTT through to most latency sensitivity tasks (level 3) implies a very short RTT. Level 1 tasks characterise an RTT in the order of milliseconds, while level 2 and 3 tasks characterise an RTT of ten-of-milliseconds and sub-milliseconds respectively. Our earlier study [34] on Cloud workload latency sensitivity has shown that most of the Cloud workloads are of least sensitivity, and the most latency sensitivity workloads are insignificant in number in comparison with the workloads of other levels of latency.

## 4 MODEL DESCRIPTION

This section describes our proposed prediction model focussed on predicting the workload arrival trend at the datacentres.

### 4.1 K-means Algorithm

James. McQueen puts forward the K-means clustering algorithm [35] based on dynamic partitioning. The basic idea of this algorithm is described as follows: Firstly, the algorithm selects a certain number of datacentres for each data catalogue in the process of dealing with data; Secondly, the data points will be divided into the respective classes based on a distance comparison between a given point and the centre of every class; Thirdly, the algorithm executes a loop iteration until the clustering criterion function is met to achieve an optimal state. This process makes data points clustered within a given catalogue relatively independent to other catalogues.

The proposed model uses K-means algorithm to cluster the workloads in respective dataset $D_n$. Firstly, $k$ data points will be selected randomly as initial cluster centres. Each data is divided into corresponding clusters based on the similarity measure, such that the data within same clusters have higher similarity and the similarity between different clusters is low. Finally, $k$ different classes are obtained. The average of each class is then calculated respectively as a new clustering centre of each class, and the above steps are repeated until the clustering centres of each class become static. The mathematical representation of the K-means algorithm is presented as follows.

For a given dataset $X = \{x_m\}$, where $m = 1, 2, ..., n$. The samples contained in the dataset $X$ are described by $m$ attributes, namely {A1, A2, ... Am}. Suppose that there are two data samples $x_i$ and $x_j$, where $x_i = (x_{i1}, x_{i2}...x_{im})$ and $x_j = (x_{j1}, x_{j2}, ...x_{jm})$, then $x_i$ and $x_j$ respectively corresponds to $m$ description attributes, and these m attributes correspond to values $x_{i1}, x_{i2}...x_{im}$ and $x_{j1}, x_{j2}...x_{jm}$. Assuming that the sample data $X$ contains a total of $k$ subcategories, the $k$ categories are $X_1, X_2...X_k$ respectively. In addition, the cluster centres corresponding to each category are $Z_1, Z_2, ...Z_k$ respectively and the corresponding samples in each class are $m_1, m_2...m_k$ respectively.

Definition 1: The Euclidean distance between each data object is expressed as in Equation 1.

$$d(x_i, x_j) = \sqrt{\textstyle\sum_{k=1}^{m}(x_{ik} - x_{jk})^2} \tag{1}$$

In the process of clustering, data similarity can be computed based on the Euclidean distance.

Definition 2: The average of all data objects in the same class can be expressed as in Equation 2.

$$Z_j = \frac{1}{m}\sum_{x \in X_j} x \tag{2}$$

Definition 3: The error square sum criterion can be expressed as in Equation 3.

$$E = \sum_{k=1}^{k}\sum_{p \in X_i} \|p - m_j\|^2 \tag{3}$$

where $p$ is the data object in its corresponding class $X_i$.

The concrete steps of the K-means clustering algorithm are expressed as follows: Suppose, $D_n = X_1, X_2, ..., X_n$ is the original data set.

1) Enter the value of $k$, which is the initial number of clusters.

2) Randomly select $k$ data objects from the data space $D_n$ as the initial cluster centre of each original class.

3) The Euclidean distance formula is used to calculate the distance between the object left in the dataset and each clustering centre, and the data objects are clustered into the nearest class.

4) Calculate the squared error sum of $k$ clusters respectively according to Definition 3 and use the criterion function to evaluate the effect of clustering.

5) After $k$ clusters are obtained, the average of all the data in each cluster is taken as the new cluster centre of the corresponding class.

6) Repeat the above steps (3), (4) and (5) until the cluster centre is fixed or the criterion function converges.

7) Finally, $k$ clusters are obtained and the algorithm is finished.

The obvious advantage of the K-means algorithm is that it is simple and easy to implement. Another notable feature of the K-means algorithm is its quicker processing. Although the K-means algorithm may not be able to obtain the global optimal solution, this effect is not known to be affecting the prediction efficiency of the proposed model. This is because of the fact that the proposed method uses K-means algorithm only for the initial processing of Cloud workload datasets, and as long as a crude initial processing dataset can be obtained within a quick time, the aforementioned effect of K-means algorithm can be nullified. Despite its advantages, K-means algorithm still characterise a few shortcomings those needs addressing. The disadvantages of K-means algorithm are summarised as follows: Firstly, it is a requirement of K-means algorithm to determine the value of k at the beginning, which implies that the dataset should be pre-estimated for classification. The value of k is difficult to determine, especially for datasets containing workloads of more complexities and dynamism. In most cases, the number of required catalogues cannot be easily determined whilst classifying the dataset. The usual approach is to continue to test different values of k until the most effective value is found. This will seriously affect the effectiveness of the K-means algorithm. Secondly, the classification result depends on the initialisation of the classification centre. Different initial values may result in different classification results. The speed of determination of the clustering centroid will determine the speed of the K-means algorithm. Thirdly, K-means algorithm is sensitive to noise. For example, given a sample with two types of data A and B, each type of data may have several points, within a very short distance. Now, adding a new dataset where the Cloud workloads characterise significant variation to those of the existing workloads a can have significant impact on the selection of the data centre. This new point is usually regarded as the noise point. Lastly, K-means is also not particularly good at classification of categories those are very close in distance.

Even though K-means algorithm has many shortcomings, its greatest strength is its minimal complexity, which means that it can process huge amounts of data within a shorter period of time. This is of great importance in the present era of data explosion. Given this, the proposed methodology adopts an improved K-means algorithm described as follows.

### 4.2 An improved K-means algorithm

In this paper, the traditional K-means algorithm is improved based on a study of Cloud workload characteristics, so that the improved K-means algorithm can process Cloud workload datasets more efficiently. With the motivation of eliminating the complexities whilst determining the $k$ value and the selection of initial cluster centre in the traditional K-means algorithm, the relationship between Cloud user behaviour and their corresponding workloads [9] are deeply analysed. Cloud workloads are categorized into respective categories based on their behavioural characteristics in terms of their latency sensitivity. The determination of the Cloud workload type makes it easier to determine the value of $k$ in the K-means algorithm, thereby overcoming the problem of the traditional K-means clustering algorithm. Furthermore, a novel method is adopted to determine the initial cluster centre as described below. Given a dataset $X = x_1, x_2, ..., x_n$ where each $x_i$ has the attributes $\{x_{i1}, x_{i2}, ..., x_{ik}\}$.

The Euclidean distance between each object is calculated as in Equation 4.

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{m}(x_{ik} - x_{jk})^2} \tag{4}$$

The local density of any object i is calculated as in Equation 5.

$$\rho_i = \sum_{j=1}^{n} \lambda(d_{ij} - d_c) \tag{5}$$

where, $d_c$ is the truncation distance among the data points. Truncation distance is a hyper-parameter which is selected by the user according to specific circumstances. $\lambda$ is a piecewise function, whose function expression is shown in equation 6.

$$\lambda(x) = \begin{cases} 0, & x \geq 0 \\ 1, & x < 0 \end{cases} \tag{6}$$

Based on the above definition, the analysis can conclude that $\rho_i$ represents the number of points of the neighbour data, and the data point i is in the range of $d_c$. After calculating the value of $\rho$ for all the data points, all the data density values are sorted in a descending order according to the value of $\rho$. In order to enhance the computation speed, a fast sorting algorithm is incorporated. During this sorting process, we introduce a new initial clustering centre of gravity selection rule. The initial cluster centre selection rule is to select the data point with the largest rank value of $\rho_{max}$ as the first cluster centre point $x_{c1}$; choose the point whose distance from $x_{c1}$ to this point equals $2d_c$ as the second cluster centre point $x_{c2}$; select the point whose distance from $x_{c1}$ and $x_{c2}$ is equal to $2d_c$ as the third cluster centre. Similarly, the kth cluster centre $x_{ck}$ can be obtained. Through the above analysis, this paper solves two key problems of the traditional K-means clustering algorithm. The specific steps of the improved K-means algorithm are described as follows:

1) Firstly, input the initial number of clusters K and the value of the truncation distance $d_c$.

2) Determine the $k$ initial cluster centres according to the Euclidean distance $d(x_i, x_j)$ and the local density $\rho_i$ using the above-mentioned new initial cluster centre selection

rule. These initial cluster centres can be represented as: $x_{c1}, x_{c2}, \ldots, x_{ck}$.

3) The Euclidean distance formula is still used to calculate the distance between the object left in the dataset and each cluster centre, and the data objects are clustered into the corresponding nearest classes.

4) Calculate the squared error sum of $k$ clusters respectively according to Definition 3, and use the criterion function to evaluate the clustering effect.

5) After $k$ clusters are obtained, the average of all the data in each cluster is taken as the new cluster centre of the corresponding cluster.

6) Repeat the above steps (3), (4) and (5) until the cluster centre becomes static or the criterion function converges.

7) $k$ clusters will be obtained, and the algorithm is finished.

### 4.3 Artificial Neural Network

Artificial neural network is a biomimetic network that mimics the working mode of human nerve cells. A certain number of neuron-like structures are interconnected to form a working network structure. BP neural network is a typical neural network which usually consists of an input layer, one or more hidden layers, and an output layer. The role of the input layer is to receive the signals transmitted by the external transmission and to transmit these signals to the corresponding intermediate layer neuron structure. During the transmission process, the signal undergoes a certain degree of change based on a weight function.

Each intermediate layer neuron will combine all the received signals and forwards them as the input signal to the next layer of neuron structure. A non-linear processing is also included in this process to enhance the expression ability of the neural network, since the linear model is usually not capable of handling linear indivisible cases. The role of the output layer is to receive the signal processed by the hidden layers and to generate the final output after processing. BP neural network uses a negative feedback to reduce the error between the output and the target value. The system successively adjusts the weights between the output and hidden layers, and between the hidden and input layers in the opposite direction to the system input, thereby achieving the purpose of reducing the network error. BP neural network usually has a strong ability to deal with non-linear relationships between data due to the addition of non-linear transformation functions.

### 4.4 BP Neural Network Architecture

Neurons are the basic building blocks of neural networks which can be seen in Fig. 1. The main features of neurons can be summarised from the literature [36], [37], [38]. In Fig. 1, $x_1, x_2, \ldots, x_n$ are defined as the input data of the neuron; $a_{i1}, a_{i2}, \ldots, a_{in}$ represent the weight factor of every input data respectively. The activation function of the neuron is represented by $g()$; $O_i$ and represent the output data and the threshold of the neurons respectively.

In Fig. 1, the output result of the neuron can be represented by the activation function, where, $O_i = g(P_i)$ and $P_i = \sum_{j=1}^{n} a_{ij} X_j - d_c$. In this formula, $X, a_i$ and $P_i$ are defined as the input values, the connection weight value for neuron $i$and the input vector of the activation function $g()$ respectively. It is worth noting that is usually defined as the 0th input value of the neuron. Therefore, the above formula can be simplified as in Equation 7, where parameter $x = -1$, and$a_{i0} = \lambda_i$.

$$P_i = \sum_{j=0}^{n} a_{ij} X_j \tag{7}$$

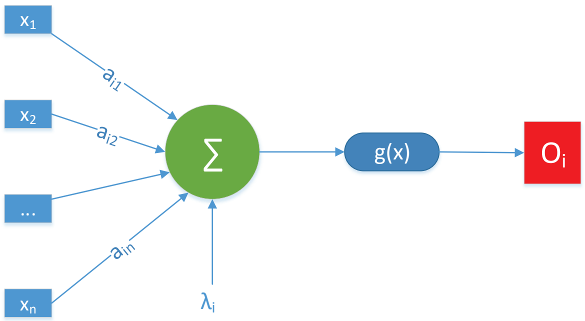


Fig. 1. Neuron Model

### 4.5 An Improved BP Neural Network Algorithm for Prediction

Forecasting models based on statistical analysis often suffer from obvious deficiencies. On the one hand, the use of such models is often accompanied by certain assumptions; on the other hand, the ability of these models to deal with complex non-linear problems is relatively weak. Faced with the complex and diverse Cloud workloads in real life, traditional prediction methods are not effective in solving such problems. The BP neural network is a representative of the artificial intelligence methodology. Each neuron in the network structure has a nuclear unit that is simple and can reflect the non-linear relationship of the data. BP neural network has the ability to reconstruct any non-linear functional relationship through the interaction of these non-linear elements.

The neural network learning process incorporates a learning law and characterise strong promotion ability, which makes the BP neural network to have a good ability to predict the future results. However, traditional BP neural networks cannot obtain prediction results quickly and efficiently because of their fixed learning rate. If the learning rate is set too large, the network cannot obtain higher precision; if the learning rate is set too low, the network cannot complete convergence within a short time. This disadvantage is especially true when the neural network needs to process large-scale data.

In order to overcome this shortcoming of the traditional neural networks, a modified BP algorithm named VLBP (variable learning rate backpropagation) has been proposed [38], as described in Algorithm 1. In comparison with the BP algorithm, the VLBP algorithm dynamically changes the learning rate of the system during the execution process, which continues to increase the learning rate in the case of reduced error to speed up the convergence of the network; conversely, it also reduces the learning rate in the case of an increased error, so that the system can maintain high learning accuracy.

However, the VLBP algorithm is susceptible to many local minima resulting from the irregular shake surface error. This phenomenon might cause the target value to fluctuate repeatedly at local minimum points. This slows down the convergence speed of the network and restrains its ability to obtain a global minimum in a limited time. Therefore, the VLBP algorithm needs to be further improved in order to achieve faster speed and accuracy whilst the processing largescale datasets. Our previous work [23] addressed this issue and proposed RVLBPNN (Rand Variable Learning rate Back Propagation Neural Network), which is inspired from the concept of genetic variation and further improves the VLBP algorithm. The prediction algorithm we propose adjusts the learning rate to a certain probability according to the trend of the MSE instead of blindly increasing or decreasing the learning rate in the prediction process. The learning rate may not be changed or multiplied by the factor $\rho$ which is greater than 1 when MSE increases beyond the set threshold$\zeta$ . Our proposed prediction algorithm is described in Algorithm 2.

With this strategy, the learning rate avoids the phenomenon of MSE continuing to decrease with slow updates near the local minima points. Simultaneously, there is also a certain probability of increasing the learning rate of the neurons. The RVLBPNN algorithm can obtain global minimum points as quickly as possible by effectively avoiding the local minimum points.

Thus, the adopted algorithm reduces the presence of local minimum points during the learning process, thereby improving the learning efficiencies of the network neurons. Although our previous work has improved the accuracy of Cloud workloads forecasting to some extent, we still noted a shortcoming that the model characterises reduced level of accuracy for workloads with limited historical traces and less periodicity. Therefore, this paper proposes a new forecasting model by incorporating the proposed improved K-means algorithm integrated with our earlier work of RVLBPNN.

**Algorithm 1** Variable Learning Backpropagation

```
1: start: Initialize weight $a, b$; threshold$\zeta$; learning rate $\eta$;
2: input data $x, y$;
3: computing the output $a$ of every hidden layer and output layer;
4: computing the error MSE between True and predicted values;
5: if MSE increases Then Increase $\eta$ else reduce $\eta$
6: end if
7: adjust the relevant connection weights: $a, b$;
8: repeat step 3,4,6 and 7;
9: Until the error accuracy is satisfied OR Achieve maximum execution steps
10: end
```

**Algorithm 2** Radom Variable Learning Backpropagation

```
1: start: Initialize weight $a, b$; threshold$\zeta$; learning rate $\eta$;
2: input data $x, y$;
3: computing the output $a$ of every hidden layer and output layer;
4: computing the error MSE between True and predicted values;
5: Generate a random number rand($u$)(0<rand($u$)<1);
6: if rand ($u$) is less than a defined value $Z$, then execute VLBP algorithm
7: else if MSE increases, then the learning rate $\eta$ is multiplied by a factor greater than 1 no matter MSE exceeding $\zeta$ or not
8: else the learning rate $\eta$ is multiplied by a factor between 0 and 1
9: end if
10: adjust the relevant connection weights: $a, b$
11: end if
12: repeat step 3,4,6 and 7;
13: Until the error accuracy is satisfied OR Achieve maximum execution steps
14: end
```

### 4.6 K-RVLBPNN Cloud Workloads Forecasting Model

Cloud workloads have the characteristics of large-scale fragmentation, which poses a great challenge to the existing state-of-the-art conventional workload forecasting models. Conventional workload forecasting schemes often fail to achieve satisfactory results in terms of both the computation speed and prediction accuracy. Therefore, this paper proposes a new Cloud workload forecasting scheme, namely K-RVLBPNN (K-means rand variable learning rate backpropagation neural network). By studying the behaviours of the Cloud service users and the characteristics of their corresponding workloads, this paper identifies that the Cloud workload arrival trend is significantly influenced by the user behaviours. Most of the existing forecasting models ignore the classification of Cloud workloads based on their latency sensitivity during the prediction process, and treats all the type of Cloud workloads in the same way. This significantly affects their prediction accuracy. This paper considers the workload classification and overcomes the drawbacks of the existing schemes, described as follows. Firstly, the improved K-means algorithm is used to cluster the sample workload dataset, and then the proposed RVLBPNN prediction algorithm is used to predict the future workload trends for each classified classes resulted by the improved K-means algorithm under different periods. Finally, the prediction obtained under different periods are averaged to deliver the prediction output. The proposed model characterise a satisfactory predictive effect, such that the improved K-means algorithm provides accurate classification and the K-RVLBPNN exploits the classified data to provide accurate prediction at a faster time-scale.

## 5 PERFORMANCE EVALUATION

### 5.1 Experiment Sample

This section demonstrates the efficiency of our proposed prediction model based on K-RVBLPNN. The dataset used in the experiments is the publically available Google workload traces [39], comprising more than 46,093,201 tasks including all the types of workloads such as CPU- intensive, memory-intensive and both CPU-intensive and memory-intensive workloads. The dataset parameters include time, job id, parent id, number of cores (CPU workloads), and memory (memory workloads). The prediction efficiencies of the proposed K-RVBLPNN prediction model are compared with Hidden Markov Model (HMM), Naïve Bayes Classifier (NBC), and our previously proposed prediction model RVLBPNN [23], all of them have also been evaluated in our earlier works [23]. Bayes model can offer better classification under less-fluctuating data samples, however Bayes model loses efficiency in a dynamic Cloud environment whilst predicting workloads showing greater fluctuations. HMM is a typical probabilistic approach that predicts the future state transition given a current state. In general, probabilistic approach may not scale well for Cloud workloads where certainty has a serious impact in decision making. RVLBPNN model has improved the prediction accuracy of Cloud workloads to some extent for sustainable datacentres. Our new proposed model is built based on the RVLBPNN model. Our proposed model has been evaluated for efficiency against the aforementioned three models, as they have been predominantly used in the context of Cloud workload trend prediction. All the four models are evaluated for their efficiencies in predicting memory and CPU intensive workloads accordingly. Prior to the training the samples as input to the K-RVBLPNN model, the Cloud workloads contained in the dataset are classified by the improved K-means algorithm, with the $k$ value is set to 4. The prediction model is trained with a set of 10 samples and to predict the next set of 10 samples, then the predicted output is compared with the actual set of successive 10 samples to evaluate the prediction accuracy.

MATLAB simulation environment provides a built-in model for RVLBPNN technique, modelling RVLBPNN as a supervised learning. The neural network is comprised of three layers. The 3-layer neural network can approximate any type of non-linear continuous function in theory. Ultimately, using 10 input nodes, 12 hidden nodes and 10 output nodes through a number of iterations for enhancing the prediction accuracy. The data samples are normalised and imploded in the interval (0, 1). ”Logsig” function is selected as the activation function of the input layer, the hidden layer and the output layer, so that the algorithm exhibits a good convergence rate. Further, variable learning rates and random variable learning rates are adopted, respectively. 100,000 workload data samples are used as the training data and another 100,000 data samples are used as the test data. The prediction accuracy is computed as the measure of correlations between the predicted and actual set of sample values.

## 5.2 Memory Workloads Estimation

Fig. 2. depicts the estimation results of K-RVLBPNN, HMM, NBC and RVLBPNN models respectively in terms of their prediction accuracy whilst predicting the memory intensive workloads. The number of experiment (X-axis) is plotted against the prediction accuracy in terms of the accuracy percentage (Y-axis) for the four models. For presenting the testing results with a better interpretation, the sample results are sorted ascendingly from 1 to 10 based on the prediction results. The average accuracy percentage in estimating the memory intensive workloads without considering the latency levels of individual workloads for NBC, HMM, RVLBPNN and K-RVLBPNN are 47.59%, 57.0%, 61.41% and 70.21%, respectively, as shown in Fig. 2 and Fig. 3. It is evident from Fig. 2 and Fig. 3 that the K-RVLBPNN exhibits a better prediction accuracy than HMM, NBC and RVLBPNN techniques. It can be depicted from the estimation results that our proposed K-RVLBPNN model is demonstrating higher prediction accuracy than HMM and NBC respectively. More importantly, K-RVLBPNN also shows a better performance (8.8% higher) than the original RVLBPNN model.

This improved prediction accuracy of the K-RVLBPNN model is attributed to the incorporation of the improved K-means algorithm for accurately classifying the workloads based on their latency sensitivity prior to the prediction process. The efficiency of our proposed model is further evaluated in forecasting memory-intensive workloads of different latency sensitivity levels. Fig. 4 depicts the estimation results of our proposed K-RVLBPNN model in terms of their prediction accuracy whilst predicting memory intensive workloads of different latency sensitivity levels as described earlier in Section IV. In addition, the prediction results are compared with our previously proposed RVLBPNN model. It can be observed from Fig. 4 that less latency sensitive memory workloads are more predictable, with the prediction accuracy being 73.30% for level 3 work-loads and 85.28 % for level 0 workloads, respectively using K-RVLBPNN. However, the prediction accuracy of RVLBPNN are 66.27% for level 3 and 77.08% for level 0. Meanwhile, the results also show that all the prediction results by K-RVLBPNN model characterise higher accuracy than those of the RVLBPNN model.

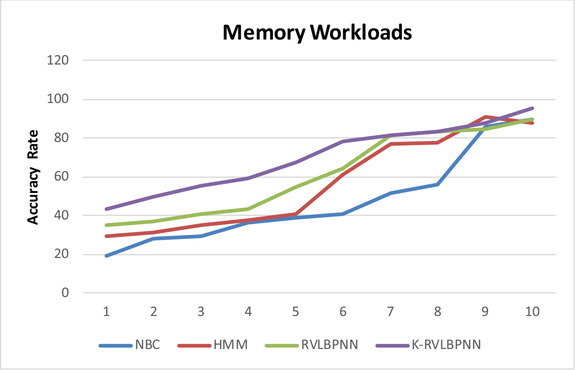


Fig. 2. Prediction of Memory-intensive Workloads

## 5.3 CPU Workloads Prediction

Similar to the memory intensive workloads, the experiments are repeated for the CPU intensive workloads from the dataset. Fig. 5 depicts the estimation results of the proposed K-RVLBPNN, RVLBPNN, HMM and NBC whist predicting the CPU intensive workloads. The average prediction accuracy of NBC, HMM, RVLBPNN and K-RVLBPNN models are 50.87%, 47.36%, 52.90% and 61.60%, respectively whilst predicting CPU intensive workloads, as shown in Fig. 6. It

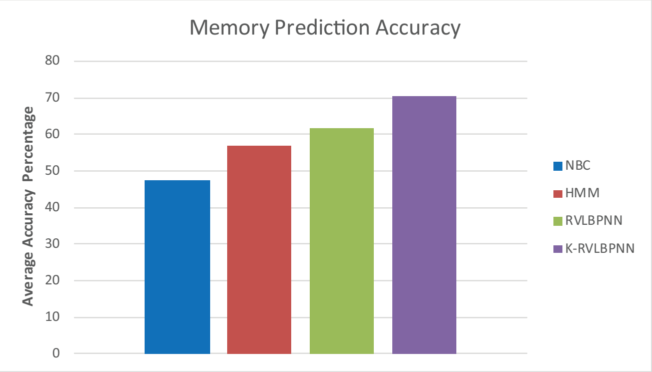


Fig. 3. Prediction of Memory-intensive Workloads

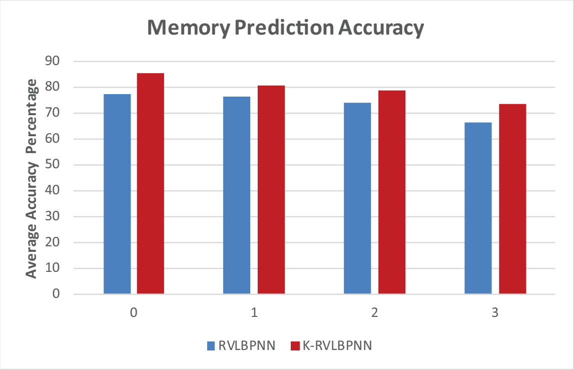


Fig. 4. Latency-wise Prediction Accuracy for Memory Workloads

can be observed that K-RVLBPNN exhibits better prediction accuracy than the HMM, NBC and RVLBPNN models by a margin of around 10.7%, 14.2% and 8.7%, respectively.

The efficiency of our proposed prediction model is further evaluated in predicting the CPU intensive workloads of different latency levels. Fig. 7 depicts the estimation results of our proposed K-RVLBPNN model and RVLBPNN model whilst predicting the CPU intensive workloads of different latency sensitivity levels. A similar trend of prediction accuracy is observed between both the memory and CPU workloads of different latency sensitivity levels. Again, CPU intensive workloads of less latency levels are exhibiting better predictability, with the accuracy of RVLBPNN being 66.17 % for level 3 workloads, 73.56% for level 2 workloads, 76.37% for level 1 workloads, and 76.88 % for level 0 workloads. Our proposed prediction model K-RVLBPNN presents higher prediction accuracy percentage. The average accuracy rates are 84.96%, 80.28%, 76.44% and 69.83% for level 0, level 1, level 2, and level 3, respectively. This leads us to infer that least-latency sensitivity workloads exhibit a better rate of prediction accuracy for both CPU and memory intensive workloads.

### 5.4 Discussion

From the experiment results, it is clearly evident that our proposed K-RVLBPNN model demonstrates better prediction accuracy than HMM, NBC and RVLBPNN models by a considerable margin. Our proposed model outperforms the other two models whilst predicting both the CPU intensive and memory intensive workloads. Meanwhile, we also observed that increasing levels of latency sensitivity of both CPU and memory intensive workloads impose increasing error margin in the prediction results. Lower level of latency sensitivity exhibits better predictability. Since the majority of the Cloud workloads are of lower latency sensitivity levels, our proposed prediction model can accurately predict the trend of most of the arriving workloads in Cloud datacentres. An increased level of intrinsic similarity among the arriving workloads facilitates a better learning rate of the neurons in the K-RVLBPNN model, which results in an increased prediction accuracy. From the experiments, we postulate that workloads should be classified based on their latency sensitivity prior to prediction to deliver better

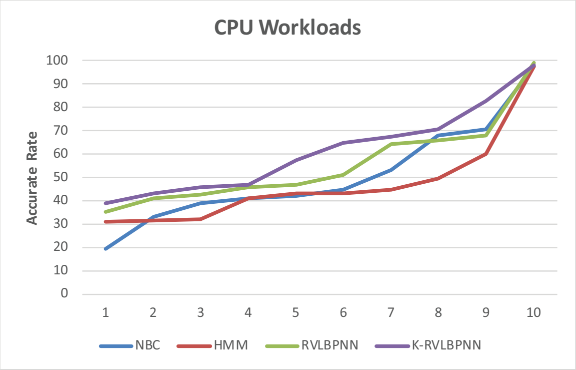


Fig. 5. Prediction of CPU-intensive Workloads

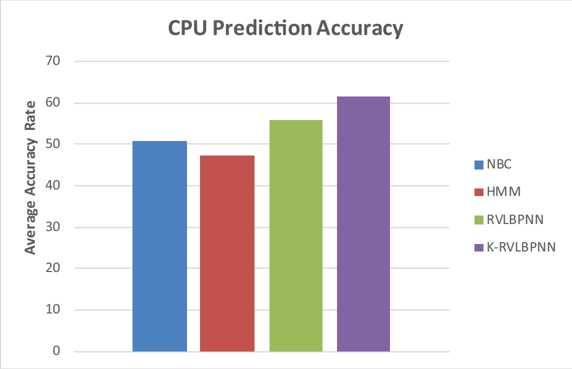


Fig. 6. Prediction Accuracy for CPU-intensive Workloads

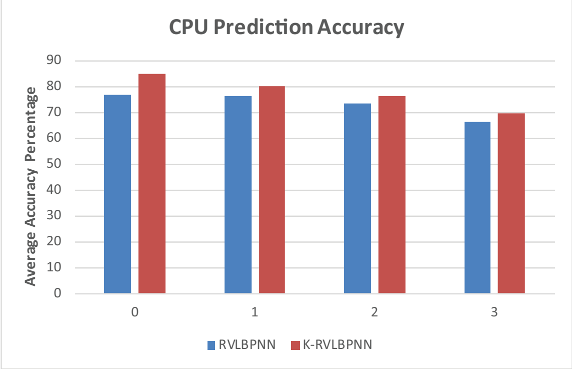


Fig. 7. Latency-wise Prediction Accuracy for CPU Workloads

accuracy.

## 6 Conclusion

Prediction analytics is gaining importance in various domains, particularly Cloud datacentres can significantly benefit from the prediction analytics of workloads. Prediction of Cloud workload behaviours and requirements can benefit resource management, server management, resource allocation and provision etc. The reliability of such prediction analytics is crucial for various reasons in a Cloud datacentre including uninterrupted services, SLA and QoS maintenance etc. This paper proposed a new workload prediction model named K-RVLBPNN, based on an improved K-means clustering algorithm and BP Neural Network algorithm. The experimental results indicate that the proposed K-RVLBPNN model achieves better prediction accuracy than the HMM-based and NBC-based prediction techniques and our earlier RVLBNN technique. Classifying the workloads based on their latency sensitivity has a significantly positive effect in the prediction process, which has been incorporated in our proposed model. Our proposed model exhibits better prediction accuracy for less latency sensitive workloads. As a future work, the possibilities of improving the prediction accuracy for higher level of latency sensitivity workloads of our proposed approach will be explored. Workloads with higher level of latency sensitivity usually have more stringent resource requirements. Resource requirements must be met within a very short time, otherwise the related tasks will fail. At the same time, such type of workloads exhibit more variable characteristics, which increases their prediction complexity. This is the reason why our model characterise a lower forecasting accuracy in comparison with the accuracy of the workloads of less latency levels. Thus, optimising our model for improving the prediction accuracy of workloads characterising higher levels of latency sensitivity is our immediate future work.

## Acknowledgment

This work was partially supported by the National Natural Science Foundation of China under Grants No. 61502209 and 61502207, Natural Science Foundation of Jiangsu Province under Grant BK20170069, and UK-Jiangsu 20-20 World Class University Initiative programme.

## References

[1] Xiangle Cheng, Yulei Wu, Geyong Min, and Albert Y Zomaya. Network function virtualization in dynamic networks: A stochastic perspective. *IEEE Journal on Selected Areas in Communications*, 2018.

[2] Hussain Al-Aqrabi, Lu Liu, Richard Hill, and Nick Antonopoulos. Cloud bi: Future of business intelligence in the cloud. *Journal of Computer and System Sciences*, 81(1):85–96, 2015.

[3] Truong Vinh Truong Duy, Yukinori Sato, and Yasushi Inoguchi. Performance evaluation of a green scheduling algorithm for energy savings in cloud computing. In *Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on*, pages 1–8. IEEE, 2010.

[4] Yulei Wu, Fei Hu, Geyong Min, and Albert Y Zomaya. *Big Data and Computational Intelligence in Networking*. CRC Press, 2017.

[5] Jianxin Li, Jieyu Zhao, Yi Li, Lei Cui, Bo Li, Lu Liu, and John Panneerselvam. imig: Toward an adaptive live migration method for kvm virtual machines. *The Computer Journal*, 58(6):1227–1242, 2015.

[6] Jianxin Li, Bo Li, Tianyu Wo, Chunming Hu, Jinpeng Huai, Lu Liu, and KP Lam. Cyberguarder: A virtualization security assurance architecture for green cloud computing. *Future Generation Computer Systems*, 28(2):379–390, 2012.

[7] John Panneerselvam, Lu Liu, Richard Hill, Yongzhao Zhan, and Weining Liu. An investigation of the effect of cloud computing on network management. In *High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems (HPCC-ICESS), 2012 IEEE 14th International Conference on*, pages 1794–1799. IEEE, 2012.

[8] Si-Yuan Jing, Shahzad Ali, Kun She, and Yi Zhong. State-of-the-art research study for green cloud computing. *The Journal of Supercomputing*, 65(1):445–468, 2013.

[9] John Panneerselvam, Lu Liu, Nick Antonopoulos, and Yuan Bo. Workload analysis for the scope of user demand prediction model evaluations in cloud environments. In *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*, pages 883–889. IEEE Computer Society, 2014.

[10] Zhitao Wan. Sub-millisecond level latency sensitive cloud computing infrastructure. In *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2010 International Congress on*, pages 1194–1197. IEEE, 2010.

[11] Hui Zhang, Guofei Jiang, Kenji Yoshihira, Haifeng Chen, and Akhilesh Saxena. Intelligent workload factoring for a hybrid cloud computing model. In *Services-I, 2009 World Conference on*, pages 701–708. IEEE, 2009.

[12] Christian Glasner and Jens Volkert. Adaps–a three-phase adaptive prediction system for the run-time of jobs based on user behaviour. *Journal of Computer and System Sciences*, 77(2):244–261, 2011.

[13] Christoph Fehling, Frank Leymann, Ralph Retter, Walter Schupeck, and Peter Arbitter. *Cloud computing patterns: fundamentals to design, build, and manage cloud applications*. Springer, 2014.

[14] Mario Bambagini, Mauro Marinoni, Hakan Aydin, and Giorgio Buttazzo. Energy-aware scheduling for real-time systems: A survey. *ACM Transactions on Embedded Computing Systems (TECS)*, 15(1):7, 2016.

[15] Jun Guo, Jing Wu, Jun Na, and Bin Zhang. A type-aware workload prediction strategy for non-stationary cloud service. In *Service-Oriented Computing and Applications (SOCA), 2017 IEEE 10th International Conference on*, pages 98–103. IEEE, 2017.

[16] Gabor Kecskemeti, Zsolt Nemeth, Attila Kertesz, and Rajiv Ranjan. Cloud workload prediction based on workflow execution time discrepancies. *arXiv preprint arXiv:1803.06924*, 2018.

[17] Niharika Verma and Anju Sharma. Workload prediction model based on supervised learning for energy efficiency in cloud. In *Communication Systems, Computing and IT Applications (CSCITA), 2017 2nd International Conference on*, pages 66–71. IEEE, 2017.

[18] Chu-Fu Wang, Wen-Yi Hung, and Chen-Shun Yang. A prediction based energy conserving resources allocation scheme for cloud computing. In *Granular Computing (GrC), 2014 IEEE International Conference on*, pages 320–324. IEEE, 2014.

[19] Asit K Mishra, Joseph L Hellerstein, Walfredo Cirne, and Chita R Das. Towards characterizing cloud backend workloads: insights from google compute clusters. *ACM SIGMETRICS Performance Evaluation Review*, 37(4):34–41, 2010.

[20] Peter A Dinda and David R O'hallaron. Host load prediction using linear models. *Cluster Computing*, 3(4):265–280, 2000.

[21] In Kee Kim, Wei Wang, Yanjun Qi, and Marty Humphrey. Cloudinsight: Utilizing a council of experts to predict future cloud application workloads. In *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, pages 41–48. IEEE, 2018.

[22] Fan-Hsun Tseng, Xiaofei Wang, Li-Der Chou, Han-Chieh Chao, and Victor CM Leung. Dynamic resource prediction and allocation for cloud data center using the multiobjective genetic algorithm. *IEEE Systems Journal*, 12(2):1688–1699, 2018.

[23] Yao Lu, John Panneerselvam, Lu Liu, and Yan Wu. Rvlbpnn: A workload forecasting model for smart cloud computing. *Scientific Programming*, 2016, 2016.

[24] John Panneerselvam, Lu Liu, and Nick Antonopoulos. An approach to optimise resource provision with energy-awareness in datacentres by combating task heterogeneity. *IEEE Transactions on Emerging Topics in Computing*, 2018.

[25] John Panneerselvam, Lu Liu, Yao Lu, and Nick Antonopoulos. An investigation into the impacts of task-level behavioural heterogeneity upon energy efficiency in cloud datacentres. *Future Generation Computer Systems*, 83:239–249, 2018.

[26] Arijit Khan, Xifeng Yan, Shu Tao, and Nikos Anerousis. Workload characterization and prediction in the cloud: A multiple time series approach. In *Network Operations and Management Symposium (NOMS), 2012 IEEE*, pages 1287–1294. IEEE, 2012.

[27] John Panneerselvam, Lu Liu, and Nick Antonopoulos. Inotrepcon: Forecasting user behavioural trend in large-scale cloud environments. *Future Generation Computer Systems*, 80:322–341, 2018.

[28] Ismael Solis Moreno, Peter Garraghan, Paul Townend, and Jie Xu. An approach for characterizing workloads in google cloud to derive realistic resource utilization models. In *Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on*, pages 49–60. IEEE, 2013.

[29] Nilabja Roy, Abhishek Dubey, and Aniruddha Gokhale. Efficient autoscaling in the cloud using predictive models for workload forecasting. In *Cloud Computing (CLOUD), 2011 IEEE International Conference on*, pages 500–507. IEEE, 2011.

[30] Zhitao Wan. Cloud computing infrastructure for latency sensitive applications. In *Communication Technology (ICCT), 2010 12th IEEE International Conference on*, pages 1399–1402. IEEE, 2010.

[31] Malvinder Singh Bali and Shivani Khurana. Effect of latency on network and end user domains in cloud computing. In *Green Computing, Communication and Conservation of Energy (ICGCE), 2013 International Conference on*, pages 777–782. IEEE, 2013.

[32] Zhitao Wan, Ping Wang, Jing Liu, and Wei Tang. Power-aware cloud computing infrastructure for latency-sensitive internet-of-things services. In *Computer Modelling and Simulation (UKSim), 2013 UKSim 15th International Conference on*, pages 617–621. IEEE, 2013.

[33] Charles Reiss, John Wilkes, and Joseph L Hellerstein. Google cluster-usage traces: format+ schema. *Google Inc., White Paper*, pages 1–14, 2011.

[34] John Panneerselvam, Lu Liu, Nick Antonopoulos, and Marcello Trovati. Latency-aware empirical analysis of the workloads for reducing excess energy consumptions at cloud datacentres. In *Service-Oriented System Engineering (SOSE), 2016 IEEE Symposium on*, pages 44–52. IEEE, 2016.

[35] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

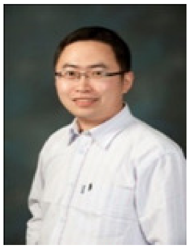
[36] Zekeriya Uykan, Cuneyt Guzelis, M Ertugrul Celebi, and Heikki N Koivo. Analysis of input-output clustering for determining centers of rbfn. *IEEE transactions on neural networks*, 11(4):851–858, 2000.

[37] Zhu Li, Qin Lei, Xue Kouying, and Zhang Xinyan. A novel bp neural network model for traffic prediction of next generation network. In *Natural Computation, 2009. ICNC'09. Fifth International Conference on*, volume 1, pages 32–38. IEEE, 2009.

[38] Martin T Hagan, Howard B Demuth, Mark H Beale, and Orlando De Jesús. *Neural network design*, volume 20. Pws Pub. Boston, 1996.

[39] Google. Google cluster data vi. 2011. https://github.com/google/cluster-data/blob/master/ClusterData2011_2.md.

**Yao Lu** received his master degree from Jiangsu University, China. He has been currently working towards the PhD degree in computer science at University of Derby, and a visiting PhD student at Jiangsu University. His current research is focused on energy efficient cloud systems and he has published his recent research works in journals, conferences and as book chapters. His research interests include Green Cloud Computing, Big Data Analytics, and High Performance Computing. He has won the best paper award in IEEE International Conference on Data Science and Systems, Exeter, 2018. He is a Member of IEEE.



**Lu Liu** is currently the Head of the Department of Electronics, Computing and Mathematics, Professor of Distributed Computing in the University of Derby, and adjunct professor at Jiangsu University. Prof. Liu received his Ph.D. degree from University of Surrey. He is the Fellow of British Computer Society and Member of IEEE. Prof. Liu's research interests are in areas of Cloud Computing, Social Computing, Data Analytics, Service-Oriented Computing and Peer-to-Peer Computing.

**John Panneerselvam** is a Lecturer in Computing at the University of Derby, United Kingdom. John received his PhD in computing from the University of Derby in 2018 and an MSc in advanced computer networks in 2013. He is an active member of IEEE and British Computer Society, and a HEA fellow. His research interests include cloud computing, fog computing, Internet of Things, big data analytics, opportunistic networking and P2P computing. He has won the best paper award in IEEE International Conference on Data Science and Systems, Exeter, 2018.

**Xiaojun Zhai** received the B.Sc. degree from the North China University of Technology, China, in 2006, and the M.Sc. degree in embedded intelligent systems and the Ph.D. degree from the University of Hertfordshire, U.K., in 2009 and 2013, respectively. He is currently a Lecturer in the School of computer Science and Electronic Engineering at the University of Essex. His research interests mainly include the design and implementation of the digital image and signal processing algorithms, custom computing using FPGAs, embedded systems and hardware/software co-design. He is a BCS member and HEA Fellow.

**Xiang Sun** received the BS degree from Jiangsu University, China, in 2013, and the MS degree from Jiangsu University, China, in 2016. He is currently working towards the PhD degree in the University of Derby, UK, a visiting PhD student at Jiangsu University. His research interests include event detection, data mining, social computing, and cloud computing.

**Nick Antonopoulos** is currently the Pro Vice Chancellor of Research and Innovation in the University of Derby and the University of Derby Technical Coordinator of the framework collaboration with CERN as well as the ALICE experiment. Nick holds a PhD in Computer Science from the University of Surrey in 2000. His research interests include Cloud Computing, P2P Computing, software agent architectures and security. Nick has over 18 years of academic experience and has published more than 150 articles in fully refereed journals and international conferences.