IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 4, 2018, accepted October 22, 2018, date of publication November 12, 2018, date of current version June 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2880694

Progress on Artificial Neural Networks for Big Data Analytics: A Survey

HARUNA CHIROMA[1], (Member, IEEE), USMAN ALI ABDULLAHI[1], SHAFI'I MUHAMMAD ABDULHAMID[2], (Member, IEEE), ALA ABDULSALAM ALAROOD[3], LUBNA A. GABRALLA[4], NADIM RANA[5], LIYANA SHUIB[6], IBRAHIM ABAKER TARGIO HASHEM[7], DADA EMMANUEL GBENGA[8], ADAMU I. ABUBAKAR[9], AKRAM M. ZEKI[9], AND TUTUT HERAWAN[10]

[1]Department of Computer Science, Federal College of Education (Technical), Gombe 60, Nigeria
[2]Department of Cyber Security Science, Federal University of Technology, Minna 234, Nigeria
[3]Department of Computer Science, Faculty of Computing and Information Technology, University of Jeddah, Raids 21432, Saudi Arabia
[4]Department of Computer Science and Information Technology, Community College, Princess Nourah Bint Abdulrahman University, Riyadh 248, Saudi Arabia
[5]College of Computer Science and Information Systems, Jazan University, Jazan 240, Saudi Arabia
[6]Department of Information Systems, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur 50603, Malaysia
[7]School of Computing and IT, Taylors University, Subang Jaya 47500, Malaysia
[8]Department of Computer Engineering, University of Maiduguri 243, Nigeria
[9]Department of Computer Science, International Islamic University of Malaya, Kuala Lumpur 53100, Malaysia
[10]Department of Information Systems, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur 50603, Malaysia

Corresponding authors: Haruna Chiroma (freedonchi@yahoo.com), Ibrahim Abaker Targio Hashem (ibrahimabaker.targiohashem@taylors.edu.my), and Adamu I. Abubakar (100adamu@gmail.com).

This work was supported by Tetfund Institutional-Based Research Grants—Federal College of Education (Technical), Gombe, Nigeria.

ABSTRACT Approximately 2.5 quintillion bytes of data are emitted on a daily basis, and this has brought the world into the era of "big data." Artificial neural networks (ANNs) are known for their effectiveness and efficiency for small datasets, and this era of big data has posed a challenge to the big data analytics using ANN. Recently, much research effort has been devoted to the application of the ANN in big data analytics and is still ongoing, although it is in it is early stages. The purpose of this paper is to summarize recent progress, challenges, and opportunities for future research. This paper presents a concise view of the state of the art, challenges, and future research opportunities regarding the applications of the ANN in big data analytics and reveals that progress has been made in this area. Our review points out the limitations of the previous approaches, the challenges in the ANN approaches in terms of their applications in big data analytics, and several ANN architecture that have not yet been explored in big data analytics and opportunities for future research. We believe that this paper can serve as a yardstick for future progress on the applications of the ANN in big data analytics as well as a starting point for new researchers with an interest in the exploration of the ANN in big data analytics.

INDEX TERMS Big data analytics, artificial neural networks, evolutionary neural network, convolutional neural network, dataset.

I. INTRODUCTION

It is estimated that 2.5 quintillion bytes of data are being produced daily; this has brought the world into the era of big data [1]. The growth of this data is continuing to increase exponentially [2]. Recent work has shown that big data has attracted unprecedented attention from both academics and industries. Massive amounts of data are being collected by many organizations on an ongoing basis. These datasets are being collected from various sources, including but not limited to the World Wide Web (WWW), social networks and sensor networks [3]. The discovery of knowledge from unstructured data accumulated from the WWW remains a difficult task because the content is suitable for human consumption rather than for machines [4].

Experimental evidence has shown that if big datasets are exploited and managed properly, it can give rise to critical intelligence that can motivate informed decisions and

The associate editor coordinating the review of this manuscript and approving it for publication was Shirui Pan.

wider vision. The challenge in the big data era is to discover knowledge from big data using new techniques that were not imagined in the past [5].

The traditional methods such as econometric, statistical and mathematical models were mainly applied for data analytics in the 1970s [6]. However, the traditional methods are only effective in solving linear or near-linear problems, and some complex nonlinear time varying problems with limitations. The limitations of the traditional methods have prompted increasing attention in computational intelligence algorithms such as artificial neural network (ANN) due to their ability to solve complex real world problems better than the more traditional methodologies [7]. The ANN and other intelligent methodologies such as genetic algorithms (GA), rough sets, support vector machines (SVM), neuro-computing, fuzzy decision trees, fuzzy logic, etc. in hybrid, ensemble or single form can be exploited to perform data analytics [8].

Most of the algorithms pointed out effectively work on sizable datasets. The emergence of big data has posed a serious challenge to the research community in terms of how best to efficiently analyze these voluminous datasets, which are generally not in a central location. [5] has shown that the flow of data in a network has changed due to the emergence of big data. In general, data in a network move from one server to another. To realise the complete potential of big data, efficient and effective algorithms that can analyse this data are required; without these, the potential of big data cannot be explored [9].

Recently, researchers have made attempts to apply ANN within the context of big data analytics. Some progress has been made in this field using ANN, despite features of big data such as high volume, velocity, variety and diversity.

In spite of the significance of the applications of ANN in big data analytics in exploring the potential of big data, we have not found a review that presents progress, challenges and future research direction on the application of ANN in big data analytics. This is to the best of the authors knowledge. However, [10] presented a survey that is limited to deep learning on big data analytics.

This paper aims to review the attempts made by the research community in exploring the potential of big data using ANN. This is to provide researchers with the state-of-the-art progress and the challenges of applying ANN within big data analytics, and to point out the potential for future research. This work also aims to identify for researchers the areas of ANN that have not been explored or which have received little attention from the research community. This review is intended for researchers to use as a standard in further exploration of the ANN that have not been explored in big data analytics. Unlike [10] the present study survey ANN applications to big data analytics in general without limitation to specific ANN architecture or model. The contributions of this work are summarised as follows:

a. We present a taxonomy of the ANN according to their architecture within the big data analytics.
b. We present a concise and precise summary of recent progress in the application of ANN to big data analytics.
c. We show that ANN is the primary computational intelligence algorithm that has facilitated the exploration and exploitation of big data's potential.
d. We demonstrate a trend in publications within big data analytics of the use of ANN, which is expected to grow rapidly in the very near future.
e. We present several powerful computational intelligence algorithms that received little attention in big data analytics to researchers for easy identification.
f. We highlight the challenges involved in previous attempts to explore big data using ANN and future research opportunities.

The rest of this paper is organised as follows: Section II presents the taxonomy and basic theories of the ANN used in big data analytics. Section III gives a description, benchmarks and examples of big data, including real world examples. Section IV presents a review of the studies that have attempted the application of ANN within big data analytics. Section V presents a discussion which includes the strengths and limitations of the previous approaches. Section VI describes future research opportunities, and concluding remarks are presented in Section VII.

II. TAXONOMY OF THE ARTIFICIAL NEURAL NETWORKS

As discussed above, progress has been made in the application of ANN within the big data analytics. In the literature, the primary algorithm that has been applied in big data analytics is the ANN.

A. TAXONOMY

The most commonly used ANN in the analysis and processing of big data is the deep neural network (DNN), also known as the deep learning or large-scale ANN. This is because the DNN has capability to process unstructured data, one of the main features of the big data [11]. There are other categories of the ANN that have made inroads into big data analytics, and these include the large-scale recurrent ANN, echo state network (ESN), back-propagation ANN (BPNN), SVM and the functional link network (FLN). These ANNs can be roughly categorised as shown in Figure I.

The ANN is one of the major soft computing algorithms and is commonly used for applications in various domains [12]. The basic theoretical concepts underlying the application in big data analytics of the different ANNs categories shown in Figure I are briefly described in the following sub-section. This aims to provide the reader with an understanding of the ways in which the various ANN functions to achieve their core objectives.

B. ECHO STATE NETWORK

Reference [13] has described the ESN as a class of recurrent ANN that can be partitioned into three segments. The $\boldsymbol{N}_{\mathbf{i}}$-dimensional input vector $x[n] \in \Re^{N_i}$ is served

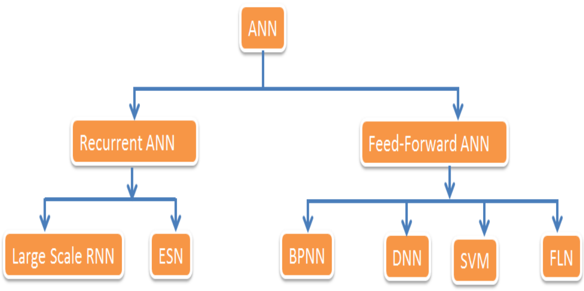


FIGURE 1. Taxonomy of ANNs based on their applications in big data analytics.

into the N_i-dimensional input reservoir with an internal state of $h[n-1] \in \Re^{N_r}$. The internal state is updated using Equation (1):

$$h[n] = f_{res}(W_i^r x[n] + W_r^r h[n-1] + W_o^r y[n-1]) \quad (1)$$

where $\boldsymbol{W}_i^r \in \boldsymbol{\Re}^{N_r \times N_i}$, $\boldsymbol{W}_r^r \in \boldsymbol{\Re}^{N_r \times N_r}$ *and* $\boldsymbol{W}_o^r \in \boldsymbol{\Re}^{N_r \times N_o}$ are matrices generated randomly, $f_{res}(\cdot)$ is a non-linear function and the previous N_o-dimensional output of the ESN is $y[n-1] \in \Re^{N_o}$. A small uniform noise may be added to the state update before the computation of $f_{res}(\cdot)$ to improve stability. Thus, the current output of the ESN is computed based on Equation (2):

$$y[n] = f_{out}(W_i^o \times [n] + W_r^o h[n]) \quad (2)$$

The $\boldsymbol{W}_i^o \in \boldsymbol{\Re}^{N_o \times N_i}$, $\boldsymbol{W}_r^o \in \boldsymbol{\Re}^{N_o \times N_r}$ are adapted according to the available training dataset and the invertible non-linear function is f_{out}. The reservoir of the ESN must satisfy the echo state property in order for the ESN to be used in learning applications. The training phase of the ESN involves feeding inputs to the network reservoir. During the training, the target output is used since there is no ESN output for feedback. The hidden matrix H and output vector d are defined as shown in Equations (3) and (4).

$$H = \begin{bmatrix} X^T[1]h^T[1] \\ \vdots \\ X^T[1]h^T[Q] \end{bmatrix} \quad (3)$$

$$d = \begin{bmatrix} f_{out}^{-1}(d[1]) \\ \vdots \\ f_{out}^{-1}(d[Q]) \end{bmatrix} \quad (4)$$

The best weight vector output of the ESN is obtained based on Equation (5):

$$w^* = \underset{w \in \Re^{N_i + N_r}}{\arg\min} \frac{1}{2} \|Hw - d\|_2^2 + \frac{\lambda}{2} \|w\|_2^2 \quad (5)$$

where the $w = [\boldsymbol{W}_i^o \boldsymbol{W}_r^o]^T$ *and* $\lambda \in \boldsymbol{\Re}^+$ is the *regularization function*. The ESN does not need the back-propagation algorithm on its recurrent component, as such, immune itself from vanishing gradient and exploding problems [14], [15].

C. DEEP NEURAL NETWORK

The DNN is a class of feed-forward ANN with multiple hidden layers. In DNN, every hidden neuron, j, uses the *logistic* function for mapping the whole input from the lower layer x_j to y_j (Equation 6), which is the scalar state transmitted to the higher layer.

$$y_j = \log istic(x_j) = \frac{1}{1 + \ell^{-x_j}}, \quad x_j = b_j + \sum_i y_j w_{ij} \quad (6)$$

where b_j, i, w_{ij} are the bias on neuron j, the index on the lower layer and the weight connection to neuron j from neuron i of the layer below. However, multi-class classification converts the total input attributes into class probability p_j by applying nonlinear "softmax", as shown in Equation (7):

$$p_j = \frac{\exp(x_j)}{\sum_k \exp(x_k)} \quad (7)$$

where k represents an index class for the whole classes. The DNN can be trained using the back-propagation based on a cost function that assesses the error between the DNN output and the target value generated for each of the training instances. For "softmax" the cost function C is a cross entropy between the actual probability d and softmax p output as computed by Equation (8):

$$C = -\sum_j d_j \log p_j \quad (8)$$

where the actual probabilities take values of 1 or 0, which are the supervised data given for the training of the DNN. For the application of DNN to a large dataset, it is found to be more efficient to perform computation with a subset of the dataset than training the complete dataset before updating the DNN weights w in proportion to the gradient [16]. The DNN has shown excellent improvement in tasks such as speech recognition, object detection and visual object recognition [17]. The DNN has different architecture. However, we only discussed the architecture we found to have been applied in big data analytics. For example, convolutional neural network (ConvNet).

D. CONVOLUTIONAL NEURAL NETWORK

The ConvNet is a DNN architecture proposed in [18]. Later, the ConvNet was revised in [19]. The architecture of the ConvNet comprised of multiple hidden layers. The hidden layers of the ConvNet includes: convolutional, pooling, and fully connected layer. Convolution is the use of weighted average operation to smoothen noisy data. The convolution performed multiple number of convolution to yield multiple number of linear activation functions. The linear activation function is run through nonlinear activation function at the second stage referred to as the detector stage. The pooling makes representation to be estimated invariant of the input small translation. Un-alteration for most of the pooled output if the inputs is translated with small amount is referred to as the invariant translation. The ConvNet possessed the capability of performing automatic feature engineering [20].

E. FUNCTIONAL LINK NETWORK

An FLN, as proposed by [21], is a higher order ANN that has no hidden neurons. The FLN has the ability to capture the non-linear relationship of input-output despite the fact that the FLN is linear in nature. That is as long as the FLN is fed with a sufficient polynomial set or functions. These may be the subset of the whole set of orthogonal basis functions that is spanning an n-dimensional representation of a space. The authentic input variables are used for the construction of the space. The FLN acts on the pattern element or the complete pattern by producing the set of autonomous linear functionality. Subsequently, these functions are appraised with the pattern as the argument. As such, separability of class is likely in the improved feature space.

Let us consider FLN with a 2 dimensional input pattern $X = [x_1\, x_2]^T$. Trigonometric function is applied to improve the pattern by expanding of the function as expressed below:

$$[x_1 \cos(\pi x_1) \sin(\pi x_1) \ldots \cos(2\pi x_1) \sin(2\pi x_1) \sin(2\pi x_1)$$
$$\ldots x_2 \cos(\pi x_2) \sin(\pi x_2) \ldots \cos(2\pi x_2) \sin(2\pi x_2) \ldots x_1 x_2]^T$$

The output of the FLN is generated by passing the weighted sum of the component of the input that is enhanced by the trigonometric through the activation function. The weight of the FLN is updated by back-propagation based on the error resulting from the FLN obtained by comparison between the FLN output and the actual value. The FLN has less computational complexity than a multi-layer ANN because the structure of the FLN comprises only a single layer [22].

F. BACK PROPAGATION NEURAL NETWORK

The BPNN is a multi-layer ANN with neurons distributed across multiple layers comprising of input, hidden (one or more) and output layers. The BPNN is fed with data through the input layer to pass it in a forward direction to the hidden layer neurons for computation with the activation function. Subsequently, it is transferred to the output layer for final computation. The error signal between the BPNN output and the target output is continuously minimised by updating the ANN weights connecting neurons in each layer of the ANN.

Let $a_1, a_2, a_3 \ldots, a_n$ be the input vectors in the input layer of an BPNN and $b_1, b_2, b_3, \ldots, b_n$ be the output vectors of the hidden layer. Therefore,

$$b_l = f\left(\sum_{h=1}^{n} V_{hi} a_h - \theta_i\right), \tag{9}$$

where $i = 1, 2, \ldots, p$, V_{hi} and θ_i are weighted between the input layer and the hidden layer and the hidden layer threshold respectively. The output layer vector is set to $c_1, c_2, \ldots, c_m$

$$c_j = f\left(\sum_{j=1}^{p} W_{ij} b_i - \theta_j\right), \tag{10}$$

where $j = 1, 2, 3, \ldots, m$, W_{ij} and θ_j weight between the hidden and output layers and threshold of the output layer. The activation function, $f(*)$ is defined as $f(x) = \frac{1}{1+e^{-x}}$. Meanwhile, the generalised error at the output units j is expressed as:

$$d_j = -\frac{\partial E_k}{\partial netc_j} = -\frac{\partial E_k}{\partial c_j} f'(netc_j) \tag{11}$$

The generalised error at the output units i is expressed as:

$$\begin{aligned} e_i &= -\frac{\partial E_k}{\partial netb_i} \\ &= -\frac{\partial E_k}{\partial b_i} f'(netb_i) \qquad (12) \\ &= f'(netb_i)\left\{-\sum_{j=1}^{q} d_j \left[\frac{\partial\left(\sum_{i=1}^{p} w_{ij} b_i\right)}{\partial b_i}\right]\right\} \\ &= f'(netb_i) \sum_{j=1}^{q} d_j w_{ij} \qquad (13) \end{aligned}$$

$$\nabla w_{ij} = -\alpha \frac{\partial E}{\partial w_{ij}} = \sum_{k=1}^{m}\left[-\alpha \frac{\partial E_k}{\partial w_{ij}}\right] \tag{14}$$

$$\nabla v_{hi} = -\beta \frac{\partial E}{\partial v_{hi}} = \sum_{k=1}^{m}\left[-\beta \frac{\partial E_k}{\partial v_{hi}}\right], \tag{15}$$

where E_k is the cost function, w_{ij} and v_{hi} are updated connection weights, and ∇w_{ij} and ∇v_{hi} in Equations (14) and (15) are the weight connections proportional to the negativity of the gradient [23].

G. LARGE SCALE RECURRENT NETWORK

The recurrent ANN is an ANN that has recurrent connections. Consider the recurrent ANN as given in Equation (16):

$$x(n) = x(x(n-1),\ u(n)) \tag{16}$$

The non-linear expansion that has memory which led to state vector is described by Equation (17)

$$x(n) = f(W_{in} u(n) + W_x(n-1)), \quad n = 1, \ldots, T \tag{17}$$

where $x(n), f(\cdot)$, $\boldsymbol{W}_{in}$ and $\boldsymbol{W}$ are the vectors of the reservoir unit activation at each time step n, the activation function at the neuron, the matrix of the input weight and the matrix of the internal context weight, respectively, and $x(n) \in \Re^{N_x}$, $\boldsymbol{W}_{in} \in \Re^{N_X \times N_u}$ and $\boldsymbol{W} \in \Re^{N_X \times N_X}$. The output of the recurrent ANN, $y(n)$ is realised as given in Equation (18)

$$y(n) = f_{out}(W_{out} x[u(n)]) \tag{18}$$

where $f_{out}(\cdot)$ and $\boldsymbol{W}_{out} \in \Re^{N_y \times N_x}$ are the non-linear functions and the network trained weights, respectively. The $x(u(n))$ transform the $u(n)$ into a high dimension $x(n)$ vector. The approach for the training of the recurrent ANN is a gradient descent based on supervised training [24]. The large-scale recurrent ANN is the parallelised of the described recurrent ANN to scale up the recurrent ANN and improve its performance [13].

TABLE 1. Commonalities and differences between the ANNs depicted in Figure I.

	Adaptability	Generalizat ion	Feedback connection	Fault tolerance	Reservoir	Multiple layer	Single layer	Immune to local minima	Convolution & Pooling
ESN	✓	✓	*	✓	✓	✓	*	*	*
RNN	✓	✓	✓	✓	*	✓	*	*	*
LSRNN	✓	✓	*	✓	*	✓	*	*	*
SVM	✓	✓	*	✓	*	✓	*	✓	*
FLN	✓	✓	*	✓	*	*	✓	*	*
BPNN	✓	✓	*	✓	*	✓	*	*	*
DNN	✓	✓	*	✓	*	✓	*	*	*
ConvNet	✓	✓		✓	*	✓	*	*	√

H. SUPPORT VECTOR MACHINE

Reference [25] introduced the theory of statistical learning inspired SVM. Unlike the BPNN, the SVM is immune from local minima. The generalised capability of SVM does not depend on the dimensional space. For example, assuming z samples of (r_i, s_i) as $i = 1, 2, \ldots, z$, the model can be formulated as the minimisation of the loss function, as shown in Equation (19):

$$z = \|k\|^2 + c.\frac{1}{z}\sum_{i=1}^{z} max\left\{|s_i - f(r_i)| - \varepsilon\right\} \tag{19}$$

where, c, and f are the tolerance error, the constant of regularisation, and the regularisation function respectively. From Equation (19), f is a function which can be approximated as shown in Equation (20):

$$f(r) = k.r + d, k, r \in R^{r'}, b \in R \tag{20}$$

The equivalent of Equation (18) in the form of constraining optimisation is expressed in [26] as follows:

$$\text{minimise } \frac{1}{2}\|k\|^2 + c.\frac{1}{z}\sum_{i=1}^{z}\left(\xi_i + \xi_i^*\right) \tag{21}$$

$$\text{subject to } ((k.r_i) + d) - s_i \leq \in +\xi_i \tag{22}$$

$$s_i - ((k.r_i) + d) \leq \in +\xi_i^*, \tag{23}$$

where

$$\xi_i, \xi_i^* \geq 0, i = 1, 2, \ldots, z. \tag{24}$$

III. COMMONALITIES AND DIFFERENCES OF THE NEURAL NETWORKS

The different types of the ANN described in Section 2 have certain similarities and differences including the architectural design, number of layers, behaviour, mode of operation, and so on. These common characteristics and differences between the ANNs are presented in Table 1.

In Table 1, the symbol $*$ indicates that the corresponding ANN does not possess the characteristic. In the following section, we present a description and examples of big data, including real world and benchmark data.

IV. DATASET

A. BIG DATA

The term 'big data' is defined differently in various works, and a universally accepted definition of big data has not yet been reached in the research community. Big data is a topic which is growing in popularity [27], and can be defined as an enormous dataset that is characterised based on the 'five V's': (1) volume (the size of the dataset is very large); (2) variety (the data set is in different forms); (3) velocity (the content of the data set is constantly adaptive); (4) veracity (the dataset has many choices or interoperation variables in a mixed analysis); (5) value (the values in the dataset are very large and the density is very low) [28]. In big data, the large volume of data depends on the application domain. For instance, a large volume in the context of wireless sensor networks differ from a large-scale volume in a cluster of computers [14].

B. BIG DATA: REAL WORLD

The definition of a dataset in a big data context must satisfy the requirements that characterise it. These requirements imply that the dataset should not be handled within a tolerable time and/or acceptable scope. That is in terms of perception, acquisition storage and processing by a traditional tool both in forms of software and hardware [28]. When these requirements are translated into the size of a dataset, the generally acceptable range is from terabytes to petabytes [29]. However, two challenges that prevent many researchers from making use of big datasets in their work are limited internet bandwidth and storage capacity [30]. Several researchers have therefore started using subsets of big datasets, which do not meet the definition of big data as described by Ji *et al.* [30]; datasets of the size of gigabytes are generally considered as 'big' datasets. Table 2 contains sample list of datasets that fit the description of the big datasets described in this section.

The first column in Table 2 shows the literature in which the datasets are either used or mentioned or referenced. The second column gives a brief description of the datasets. The third and fourth columns give the sizes of the big datasets and the domains from which the datasets are extracted. The datasets are all solely owned by the enterprises that generate the dataset. The datasets are not freely available,

TABLE 2. Samples of real world big datasets.

Reference	Dataset Description	Size	Domain
[31]	Walmart databases	2.5 PB	Retail
[32]	CERN Data Center in Meyrin	45PB	Scientific/Physics
[33]	Facebook Traces from internal log database of 3000 nodes cluster queried for 1 ½ month	1.5 EB	Social media
[34]	Ebay 532 nodes Hadoop cluster	5.3PB	e-Commerce
[33]	Cloudera Customer Telecommunication Traces on 300 node cluster for 9 days	600TB	Telecommunication
[35]	Global Eye surveillance system shanghai video accumulation in a Month	1.9 TB	Surveillance
[36]	Couple Model Intercomparison project 5 with over 40 models	6 PB	Climate
[37]	LinkedIn activity data compressed in Kafka clusters	100TB	Social Media
[38]	Internet archive crawls and save web pages on Wayback Machine	80 TB	Internet Archive
[39]	1000 Genomes Project containing a detailed map of human genetic variations	200 TB	Geographic/Medicine
[40]	Rainfall dataset for 4kmx4km resolution, United States	98GB	Weather
[41]	Google Clusterdata-2011_trace containing compressed 29 day's cell information on 12.5K cluster	41GB	Information Technology
[42]	Google Web1T-5gram consisting of N-gram words used for building language model	24.4GB	Language processing

having access to the datasets (Table 2) requires a subscription except [38], [39], which are freely available for researchers.

C. BIG DATA: BENCHMARKS

Benchmarks are sets of standard(s) that serve as points of reference for performance or quality level evaluations [43]. In the context of big data, benchmarks are quality measurements of products in terms of technology, tools and their comparisons with standard ones or similar ones from peer producers. The benchmarks have to include the diversity of data and workloads in an end-to-end manner, that characterise the broader use of big data systems [44]. The aspects of big data benchmarking that need to be distinguished from other types of benchmarks are the benchmarking properties, benchmark process, data generation, hardware and software of workloads [45].

TABLE 3. Samples of big data benchmarks.

Reference	Benchmark Description	Data Scalability (volume, Veracity)	Type/ No of Real data sets modeled
[46]	GraySort developed by Sort Benchmark	Not Available (N/A)	None
[47, 48]	GridMix developed by Apache!	N/A	None
[49, 50]	PigMix developed by Apache!	N/A	None
[51]	YCSB developed by Yahoo!	N/A	None
[52, 53]	AMP Benchmarks developed by amp Lab	N/A	None
[54]	HiBench developed by Intel	Partial	Unstructured text data/1
[45]	BigBench a proposal by Teradata	N/A	None
[48]	LinkBench developed by Facebook	Partial	Unstructured graph data /1
[50]	CloudSuit	Partial	Unstructured text data/1
[44]	BigDataBench developed by Chinese Academy of Science	Total	Unstructured text data / 1 Semi-structured text data/1 Unstructured graph data/2 Structured table data/1 Semi-Structured table data/1

The data generation aspect of the benchmarks is usually a tool that is developed using a certain mathematical model, careful study and thorough observation of real world datasets. The tools used for generating datasets are similar to the tools modeled when run by a user on a system and by specifying the size of the data required. The development of the big data benchmarks is still in its early stages, with the majority of the existing ones focused on either one or subset component of the big dataset. Examples of these are GraySort, GridMix, PigMix and YCSB [45]. However, there are other benchmarks that cover a broader big data system; example includes BigBench, BigDataBench and TPC-BD. Table 3 shows samples of these big data benchmarks.

In Table 3, the first, second, third and fourth columns present the reference, the description of the benchmark and it is developers, the scalability of the benchmark in terms

TABLE 4. Samples of small datasets.

Reference	Benchmark Data set	Size
[56]	BCW	121K
[57]	MAD	443K
[58]	Iris Dataset	4.4K
[59]	Letter recognition	696K
[60]	Sonar Dataset	86K
[61]	SCCTS	282K
[62]	KDD	1.3M
[63]	Dermatology Dataset	25K
[64]	Thyroid diseases dataset	754K
[65]	AID	3.8M
[66]	SRHAPTD	76M
[67]	HHAR	741M

of volume and ability to generate huge data in the range of several terabytes, and veracity, representing the uncertainty of the data, respectively. In Table 3 'Not available (N/A)' indicates that the information is not given, and 'None' signifies that no real dataset is modelled. Hence, the dataset generated is arbitrary.

D. DATASETS FROM THE UCI REPOSITORY

The UCI repository benchmark datasets are the state-of-the-art datasets commonly used by researchers in evaluating the effectiveness of a propose ANN. However, advancements in technology have started to relegate the state-of-the-art datasets. This is because the advent of big data prompts new challenges to the research community on how to apply the ANN in big data analytics. The ANN performs well on small datasets such as those in the UCI repository. Table 4 shows several sample benchmark datasets from the UCI repository frequently used for evaluating the performance of algorithms [55].

The examples of the data sets are Breast Cancer Wisconsin (BCW), the MADELON artificial dataset (MAD), also known as MADELON, the Iris dataset, letter recognition, sonar signal, adult income data set (AID), the synthetic control charts time series (SCCTS), knowledge discovery in databases (KDD), dermatology, thyroid diseases, smartphone-based recognition of human activities, postural transitions and heterogeneity activity recognition, smartphone-based recognition of human activities and postural transitions (SRHAPTD), and heterogeneity human activity recognition (HHAR). The sizes of the datasets are summarised in Table 4. A comparison of the sizes of the big datasets in Table 2 and the UCI repository data sets in Table 4 shows that there is a huge difference. The advent of big datasets has posed a challenge to the research community in terms of modifying existing algorithms or proposing new algorithms that are capable of handling big data analytics to explore the potential in big datasets.

V. THE APPLICATIONS OF ARTIFICIAL NEURAL NETWORKS IN BIG DATA ANALYTICS

Many of the existing intelligence algorithms lack sufficient scalability to deal with big datasets. Therefore, in many situations, the algorithms ended up not exploring and exploiting big datasets. A large challenge to researchers/green engineers is presented when existing algorithms fail to deal with big datasets that are produced as a result of technological advancements [68].

Despite these challenges, many research efforts into the applications of ANN in big data analytics have been attempted and are still ongoing. Though, still in their early stages. As described earlier, ANN has made inroads into big data analytics. In this section, we review the studies that have attempted the application of ANN within the context of big data analytics. The studies selected here are those that apply the ANN within big datasets and that have satisfied the definition of big datasets in Section IV. These studies are reviewed according to Figure I as follows:

A. RECURRENT NEURAL NETWORK

The RNN has been found be to applied in big data analytics, for example, instead of the commonly used centralised RNN, [14] proposed a class of decentralised RNN which is referred to as ESN since it has received little attention from researchers. ESN is based on the direction method of multipliers optimisation procedure. The ESN is experimentally tested on four benchmark big datasets. The results show that the proposed ESN has the potential to solve the problem of large and decentralised data, and can be applied in large-scale prediction and decentralised classification. The implementation of ESN on big data shows that the performance of the ESN in terms of convergence speed, accuracy and efficiency is comparable to the centralised RNN application. [11] have implemented a large-scale recurrent ANN to explore the computational power of graphical processing units (GPUs) for big data analytics during the training phase. This is to accelerate the training process of the large-scale RNNs so as to improve its performance. Experiments indicate that the time consumption required for the implementation of a large-scale recurrent ANN on the GPU is lower than for implementation on the CPU. The ANN-based Memristor is implemented in the operational phase of the large-scale ANN and is shown to improve energy efficiency.

B. FEED-FORWARD NEURAL NETWORK

The applications of the feed-forward ANN in big data analytics is discussed in this section. [9] proposed a hybrid of the similarity search and the Levenberg-Marquardt (LM) algorithm, with improved BPNN and double hidden layers (SLMDBP), which is capable of processing hydrological big data. A second hidden layer is added to the ANN to improve the ability of nonlinear mapping. A similarity search is used to search for sequences that match those in the preceding years, to avoid the problem of excessive preceding historic data.

SLMDBP is applied to predict the water level of Xingzi Station. The results indicate that the SLMDBP performs better than the BPNN and LM algorithm improved BPNN with double hidden layers (LMDBP) in terms of both accuracy and convergence speed. [69] have proposed a self-organised Kohonen network (SONN) to reduce the dimensionality of hydrographic big data so as to improve its clustering analytic. The training of the SONN is carried out in phases, as the training cannot be performed in a single phase due to limitations associated with the SONN in handling higher density data. It was found from experiment that SONN is good for clustering big data. However, SONN is not compared with other methods for evaluation purposes, and it is therefore difficult to measure the effectiveness of the SONN in clustering big data.

Reference [70] propose the visualization of traffic big data based on the fusion of SONN and graph theory (SONNGT) to guide city planning and management. The SOFMGT is realized from the fusion of self-organizing feature map neural network and graph theory. The SONNGT is able to extract features, cluster high dimensional congestion data and classified traffic congestion model.

The ability of ANN to model non-linear complex problem motivated [71] to propose a 2 level ANN model with multiple hidden layers for the prediction of dependent failure patterns of big data jobs and task. A two-level ANN model is developed by connecting two ANNs models. The first ANN model predict dependent patterns of the big data jobs, task and resources consumption of fail tasks is reduced. The second ANN model predict classes of fail jobs. The model is evaluated on Google cluster trace. Result indicated that the ANN model is able to capture failure at big data cluster, save resources of the system, reduces CPU, disk and memory consumption.

Reference [72] designed and developed big data ANN toolkit for handling the challenges of large scale biomedical text classification. The Word2vector ANN has the ability to convert word into vector. The Word2vector ANN is implemented on the big data ANN toolkit for biomedical text classification and its performance is evaluated. It is found that the Word2vector ANN performs better than the SVM, decision tree (DT), NB and LR in terms classification accuracy. [73] proposes SVM for the selection of optimal communication rule in a big data environment. The SVM is applied in the big data system to select the best communication rule. The results indicated that the SVM has the ability to select the optimal communication rule in a big data environment.

Reference [74] propose ANN to identify game playing player assiduity. The big data set used for the study was world of Warcraft game. The ANN is applied for the identification of the player assiduity using the Hadoop and visual tools on the cloud cluster. It was found that the possibility of identifying players' assiduity is 91%. It further revealed that it is possible to map and identify the behaviour and game playing pattern of a player.

Reference [75] proposes re-training for supervise learning of ANN with optimised parameters for the classification of electricity dataset for monitoring large scale data dynamically. The ANN configuration with the best parameters was searched for the segment of the data stream, to improve convergence time, loss function was used. The ANN is applied to classify Australia electricity market dataset. The result indicated that ANN is able to accurate classify the electricity dataset.

Reference [76] propose a parallel BPNN (PBPNN) based on data separation in Hadoop, HaLoop and Spark to improve accuracy and convergence speed. The data separation is executed by Map and FlatMap while the Join and Reduce executed the ensemble operations. The propose algorithm was initially evaluated on small cluster before deploying it on commercial cloud computing environment for further evaluation. Experimental result indicated that the accuracy, efficiency and stability of PBPNN outperforms the BPNN. Reference [3] parallelised ANN based on MapReduce for application within big data to improve convergence speed and classification accuracy. This is because ANNs are known for their slow convergence, especially on large datasets. The parallelisation is based on MapReduce, rather than message passing interface (MPI). This is because the MPI used in [68] for parallelising ANN is susceptible to fault intolerance. The parallelisation of the ANN based on MapReduce is done in three cases: volume of the data, size of the training data and neurons in the ANN. The experimental results indicate that the parallel BPNN is more effective than the BPNN.

The challenges of managing biomedical big data, such as in a very high feature space, imbalance distribution and sparse attributes using FLN, motivated [79] to propose a novel FLN. The proposed FLN is based on propensity score-matching methods within MapReduce and Newton Raphson maximum-likelihood optimisations (FNPNM) to deviate from the limitations of the conventional FLN in handling big data and to enhance it is performance. The FNPNM is applied for classification in biomedical big data. The experimental results indicated that the FNPNM outperforms SVM, logistic regression (LR), K-nearest neighbor (KNN), radial basis function (RBF), feed-forward ANN (FFNN) in both classification accuracy and computational time.

Reference [81] proposed a customised parallel BPNN (cNeural) for parallelising big datasets rather than the BPNN nodes, to minimise communication costs. This is in order to accelerate the training of the large-scale BPNN. The cNeural system applied HBase for the large-scale training of the big dataset storage as well as parallel loading. To speed up the cNeural training, inbuilt memory is provided. Instant messaging is provided by compact and event-driven messaging communication. Evidence based on simulation shows that the cNeural cost in terms of time and messaging communication is approximately 50 times lower than the solution based on Hadoop MapReduce. Reference [77] propose a computerised method based on BPNN for evaluating radiotherapy and chemotherapy big data to provide standard treatment and

TABLE 5. Summary of published papers that have attempted big data analytics using artificial neural network.

Reference	Dataset	Proposed ANN	Compared algorithm	Contribution/s
[11]	Big data (not specified)	Large scale RNN and GPU, ANN based Memristor	Large scale RNN, CPU, and Memristor	The effectiveness of large scale RNN on the GPU is better than on the CPU. The ANN based Memristor is found to be more energy efficient than the conventional Memristor.
[9]	Hydrological big data	SLMDBP	LMDBP and BPNN	The SLMDBP converges to optimal solution faster than the LMDBP and BP.
[69]	hydrographic big data	SONN	Not compared	SONN is good for clustering of big data.
[77]	Radiotherapy and chemotherapy big data	ANN	Not compared	The ANN is able to grade new cases.
[14]	Artificially generated large scale big data	ESN	Recurrent ANN	The performance of ESN indicated that the convergence speed, accuracy and efficiency is comparable with the fully centralized RNN.
[3]	Iris dataset, synthetic dataset	BPNN	BPNN	The experimental result indicated that the parallel BPNN is more effective than the BPNN.
[78]	Weather big data	DNN + SVR	SVR	The DNN + SVR forecast temperature better than the SVR
[79]	Biomedical big data	FNPNM	SMV, KNN, FFNN and LR	The classification accuracy and computational time of the FNPNM are found to be better than the SMV, KNN, FFNN and LR.
[80]	Speech recognition big data	DNN on IBM Blue Gene/Q	DNN on Intel Xeon	The Blue Gene/Q scales has linearly up to 4096 processes without losing accuracy.
[81]	Handwritten big data	cNeural	Hadoop MapReduce	The solution based on cNeural is almost 50 times faster than the solution emanated from Hadoop MapReduce.
[82]	20-Newsgroup and BBC news document	DDNN	DBN, SVM, KNN and NB	The DDNN improve text classification performance over DBN, SVM, KNN and NB.
[74]	World of warcraft game.	ANN	Not compared	It is possible for the ANN to map and identify the behaviour and game playing pattern of a player.
[75]	Australia electricity	ANN	Not compared	The ANN was found to accurately classify electricity dataset
[83]	Images	Parallel BPNN-PSO	BPNN-PSO and BPNN	Parallel BPNN-PSO performs better than both the BPNN-PSO and BPNN in terms of speed and accuracy
[84]	Traffic flow dataset	RBFN-MABC	KNN, RBFN, IBPNN and RBFNcGA	The RBFN-MABC improve the accuracy and run time of traffic flow prediction over the baseline algorithms
[85]	Bogies dataset from high speed train	DNN	Single hidden layered BPNN, multi hidden layered BPNN and GA-BPNN	The DNN was found to be better than the compared algorithms.
[76]	IRIS	PBPNN	BPNN	Experimental result indicated that the accuracy, efficiency and stability of PBPNN outperforms the BPNN
[86]	Large scale power system dataset	ConvNet	Feed-Forward NN	The ConvNet performs better than the Feed-Forward NN
[71]	Google trace with rich heterogeneous workloads	Two-level ANN with hidden layer	Not compared	The ANN model is able to capture failure at big data cluster, save resources, reduces CPU, disk and memory consumption.

TABLE 5. ***(Continued.)*** **Summary of published papers that have attempted big data analytics using artificial neural network.**

[87]	Healthcare post on Facebook	DNN	ANN	The DNN perform better than the ANN in predicting engagement on healthcare post on Facebook.
[72]	Medical dataset	Word2vector ANN	SVM, DT, NB and LR	Word2vector ANN performs better than the SVM, DT, NB and LR in classifying biomedical text.
[88]	Multimodal brain tumor image segmentation	D ConvNet	Not applicable	The D ConvNet is able to perform brain tumor segmentation with excellent results based on dice score and accuracy.
[89]	Labelled generated dataset	ConvNet	Bayes factor, KS distance and BIC	The ConvNet perform better than the classical methods.
[90]	Text document	R ConvNet and improved MLP	MLP	The improved pseudocode MLP was found to outperform the baseline pseudocode MLP.
[91]	Brain tumor image	Deep ConvNet	Published results	Deep ConvNet improved on the already published results.
[70]	Traffic	SONNGT	Not applicable	The SONNGT is able to extract features, cluster high dimensional congestion data and classified traffic congestion model.
[92]	Train large scale data	SELM and DELM	Classical TDPS	The SELM and the DELM were able to improve the accuracy of the classical TDPS
[93]	Large scale facial data	FD- ConvNet and PBD-ConvNet	VCC16 RCNN and DDFD	The FD- ConvNet and PBD- ConvNet are found to perform better than the VCC16 RCNN and DDFD on large scale real-world dataset detection of faces and facial features.
[94]	Image	KDA-RBF-GA	PCA + LDA+ RBF, KSVD + LDA+RBF and LDA + RBF	The KDA-RBF-GA is found to be better than the compared algorithms on most of the datasets.

improving tumor treatment. The dimensionality of the big data is reduced using principal component analysis, and the BPNN is trained based on radiotherapy and chemotherapy data to create a computerised method of evaluation. The BPNN is applied to the grading of new cases for their quantitative control score. The proposed method is not compared with other methods to ascertain its effectiveness. Table V presents a summary of the publications that have applied ANN within big data analytics.

C. DEEP NEURAL NETWORK

The DNN attracted attention of researchers for application in big data analytics. Different DNN architecture were found to be applied in the big data analytics. For example, [78] used DNN and big data from weather applications for the forecasting of hourly temperatures. The DNN used a layer-wise unsupervised pre-training mechanism in order to avoid the difficulties in training the DNN. A support vector regression (SVR) is used at the top layer of the DNN to produce the forecast value of the hourly temperature. The DNN with SVR at the top layer was found to be better than the conventional SVR in forecasting hourly temperatures. [82] propose text classifier based on denoising deep neural network (DDNN) to improve the accuracy of big text data classification. The dataset collected for the study were 20-Newsgroup and BBC news documents. The denoising auto encoder and restricted Boltzmann machine are used for denoising and feature extraction during the data cleaning phase. The denoise dataset and DNN is used to build the DDNN classifier. The DDNN model is applied for the text classification and it was found to be better than the SVM, deep belief network (DBN), k-nearest neighbor (KNN) and Naïve Bayes (NB) in text classification.

Reference [87] propose the used of DNN and ANN to predict the popularity of healthcare post on Facebook. The DNN with two and five hidden layers as well as ANN were used for the prediction of engagement on Facebook healthcare post. It was found that deeper ANN has no contribution to the prediction accuracy and is time consuming with the healthcare dataset extracted from the Facebook healthcare post. However, evidence shows that the DNN perform better than the ANN.

Reference [93] propose the lightweight DConvNet (LDCNN) to improve the performance of DConvNet in facial detection. Progressive positive and hard negative sample mining is proposed to train the LDConvNet without compromising its efficacy and improve convergence speed and accuracy. The authors put forward two ConvNets for detecting face (FD-ConvNet) and part based detection (PBD-ConvNet), respectively. The FD-ConvNet and PBD-ConvNet are found to perform better than the VCC16 RCNN and DDFD on large scale real-world dataset detection of faces and facial features.

Reference [89] proposes ConvNet for the automation of statistical analysis process for application in big data analytics. The ConvNet was constructed to perform model selection and estimation of parameter automatically. These ConvNets are referred to as the neural model selector and neural model estimator were trained using labelled data generated from the models. Results shows that the neural model selector and neural model estimator performs excellently compared to Bayes factor, Kolmogorov-Smirnov (KS) distance and Bayesian Information Criterion (BIC).

Reference [88] constructed a 6 layer dense ConvNet (DConvNet) for the segmentation of brain tumor. The DConvNet connects each of the layer to every subsequent layer of the DConvNet architecture to ensure maximum information flow. The DConvNet is applied to perform brain tumor segmentation based on Multimodal Brain Tumor Image Segmentation Challenge dataset. The DConvNet is able to perform brain tumor segmentation with excellent results based on dice score and accuracy. [91] propose a deep ConvNet to segment and classify brain tumor based on big data analytics. The architecture of the deep ConvNet consisted of 7 layers. The deep ConvNet is applied on large scale datasets for experimentation. Result indicated that the deep ConvNet is able to segment and classified the brain tumor with excellent results compared to already published results. [92] proposes shallow extreme learning machine (SELM) and deep extreme learning machine (DELM) for predicting train delay in the perspective of big data analytics to improve the accuracy of train delay prediction. The SELM and DELM were used on the large scale train data to predict the train delay and compared their performance with the state of the art train delay predictive system (TDPS). It is found that the SELM and the DELM were able to improve the accuracy of the classical TDPS. [90] propose recurrent ConvNet (RConvNet) and multi-layer perceptron (MLP) to develop artificial intelligence search engine for improving search results. The MLP is applied to detect pseudocode whereas the RConvNet to detect target text line. The RConvNet conveyed information about the algorithm efficacy. The MLP was trained with 15 features extracted from the document to improve the performance of the pseudocode MLP. The improved pseudocode MLP was found to outperform the baseline pseudocode MLP.

Reference [85] propose DNN for recognising fault in bogies. The data for the study were extracted from multibody dynamic software SIMPACK. The DNN is applied to detect fault in bogies. The fault detection accuracy and run time of the DNN is compared with single hidden layered BPNN, multi hidden layered BPNN, GA-BPNN. The DNN performs better than the compared algorithms. [86] propose ConvNet architecture to perform multi-class classification of power system for controlling complex power grid. The ConvNet is used to develop an intelligent system with capability to control complex power grids. The result indicated that the ConvNet performs better than the Feed-Forward Neural Network in terms of classification accuracy. [80] proposes DNN training based on a data-parallel Hessian-free second-order optimisation algorithm to parallelise the DNN operation in order to speed up convergence time. The DNN is explored on the Blue Gene/Q architecture for speech recognition applications within big data. The experiments conducted on large-scale speech recognition tasks indicate that the performance scales have linearly up to 4096 processes without losing accuracy.

D. SWARM AND EVOLUTIONARY NEURAL NETWORK

Few number of studies were found to apply swarm and evolutionary algorithms for the optimization of ANN parameters. The ANN optimized through swarm and evolutionary algorithms were applied in big data analytics. For example, [83] parallelizes BPNN optimized with particle swarm optimization (parallel BPNN-PSO) to improve classification accuracy and run time. The PSO optimize the threshold and initial weights of the ANN based on MapReduce on the Hadoop platform. The propose parallel BPNN-PSO is applied on big dataset for processing. The experimental results show that the parallel BPNN-PSO improve classification accuracy and convergence time over BPNN and BPNN-PSO.

Reference [84] propose the modification of artificial bee colony (ABC) for optimizing radial basis function network (RBFN) to improve prediction accuracy and run time. To improve the searching strategy of the ABC onlookers, the mutation and crossover of the differential evolutionary were used for search in the ABC instead of the original search strategy of the bees. The modified ABC is applied to optimized the weights and threshold of the RBFN (RBFN-MABC). The RBFN-ABC is applied to predict traffic flow in a big data environment. Result indicated that the performance of RBFN-MABC in terms of accuracy and run time is better than the K nearest neighbour (KNN), RBFN, improved BPNN (IBPNN) and RBFN based on cloud genetic algorithm (GA) (RBFcGA).

Reference [94] propose the use of kernel discriminant analysis (KDA) for feature extraction from high dimensional big data before applying RBF network. The parameters of the RBF network were optimized using GA (KDA-RBF-GA) and evaluated over multiple datasets to solve classification problems. The propose KDA-RBF-GA is found to be better than the face recognition algorithm based on sparse representation and RBF (KSVD + LDA + RBF), Linear Discriminant Analysis (LDA) + RBF and LDA + principal component analysis (PCA) + RBF on most of the datasets.

VI. DISCUSSION

Progress has been made in the application of ANN in big data analytics. The ANN has been found to make inroads into big data analytics.

A. STRENGTHS AND PUBLICATION TREND

Currently, the Hessian-free second-order optimisation approach coupled with the BG/Q offers the fastest training approach and the optimum WER for cross entropy and sequence training objective functions on the DNN [80]. The review indicates that the DNN is an effective algorithm for big data analytics. It is also shown that the clustering of big data can be achieved with SONN. One of the main features of big data is it is decentralised nature. The ESN has the potential

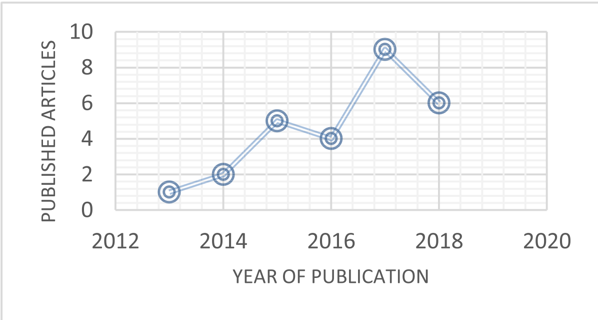


FIGURE 2. Trend in publications on big data analytics using ANN.

for making predictions and decentralised classifications using big datasets.

Figure II shows that research in the application of ANN in big data analytics is gaining attention. The first publication appeared in 2013, and the number of the publications has continued to rise each year up to 2018. Research in the application of ANN within big data is expected to continue to grow in the near future. This is predicted because unprecedented research attention presently focused on big data analytics. In addition, there is a quest for modified or new ANN that can effectively explore big datasets.

B. CHALLENGES

1) ARTIFICIAL NEURAL NETWORKS

The state-of-the-art literature on the applications of ANN in big datasets shows that the most commonly applied algorithms are the DNN and ConvNet. However, these classes of ANN are susceptible to limitations as follows: the DNN is mostly trained using first-order stochastic gradient descent, e.g. the backpropagation training algorithm [17], which is difficult to parallelise due to its serial execution; the high number of parameters involved in the training of DNN and ConvNet on big data; slow convergence speed, especially for big data; high computational cost; vanishing of gradients; exploding; and the problem of inter-processor communication cost and bottlenecks incurred by the parallelisation algorithms. Previous studies using DNN for big data analytics indicate that the researchers mainly rely on the backpropagation algorithm for the training of the DNN for big data analytics. However, the backpropagation algorithm is susceptible to falls in local minima. This also has a slow velocity for convergence to the optimum solution during learning. Currently, the long convergence time of the DNN limits the use of DNN to high-cost servers or platforms with multiple GPU cores, due to the complex structure of the DNN [95]. [3] have argued that in spite of the potential in the complete distributed BPNN in a cluster of computers, the continued initialisation and stops in MapReduce in Hadoop typically involve a high computational cost. Applications of deep RNN and spiking ANN are scarce in big data analytics despite their excellent performance in other domains [96], [97].

2) SWARM AND EVOLUTIONARY NEURAL NETWORKS

Evolutionary algorithms have been proven to be effective and efficient on small datasets and benchmark functions. However, the exploration of the evolutionary algorithms on the big datasets is less explored because of their inability to deal with big data sets. The application of evolutionary algorithms in solving problems has been successful in many domains [98]. Despite the fact that the training of ANN using evolutionary algorithms is more effective and efficient than backpropagation algorithms [99]–[101], their application to train ANN for exploring big datasets does not received tremendous attention, only limited number were found. The review shows that evolutionary algorithms have not been applied by previous researchers for the determination of weights, biases, structures, the adaptation of learning rules and internal parameters of the DNN, ESN, FLN, large-scale RNN and ConvNet within big data analytics. These may offer improved convergence speeds and accuracy. Recently, many swarm and evolutionary algorithms have emerged: the cuckoo search, flower pollination algorithm, chicken swarm optimisation, bat algorithm, artificial immune system, biogeography-based optimisation, ecogeography-based optimisation algorithms, and many other hundreds of swarm and evolutionary algorithms can be found in [102] and [103]. These swarm and evolutionary algorithms have been shown to be effective and efficient on small datasets. However, the applications of their modification, hybrid or ensemble with ANN in big data analytics require extensive studies.

3) COMPUTATIONAL COST AND ENERGY EFFICIENCY

Typically, the training of large-scale datasets requires thousands of iterations or epochs, making the process computationally expensive. Training of ANN on the big data consumed much time before convergence to the optimal solution. The training phase typically requires large volume of memory and computing resources. It is obvious that the application of the ANN in big data requires high computational cost and energy consumption. The consumption of energy contributes to the emissions of carbon dioxide (CO_2). It has already been estimated that the world information and communication technology industry has contributed 830 Mt to global CO_2 emissions. This situation have contributed a 2% increase to global warming, as argued by Pettey [104].

Big data analytics using ANN will likely continue to add to emissions of CO_2 if measures to combat this menace are not taken, due to the energy consumption required for the processing of big data using ANN. These are major challenges to the research community. Despite these challenges, very few of the researchers that have attempted to apply ANN in big data have considered reducing energy consumption and computational cost in their study. The overwhelming majority of the studies have ignored the issue of energy efficiency. One of the greatest challenges in big data analytics using ANN is energy consumption. Park *et al.* [95] have stated that energy efficient DNN is imperative for saving energy.

4) LIMITATION OF THE SMALL DATA SET

The benchmark datasets mostly used by researchers to evaluate a newly or modified ANN are not big data sets as shown in Table 4. As such, the effectiveness of the ANN is limited to a small data sets, whereas the modern reality is the big data (see Table 2). This is because of advancements in technology that have revolutionised the information and communication industry, which has led to the unprecedented accumulation of data and given birth to big data era. For example, the difficulties encountered in the applications of ANN in biomedical big data involve the complexity of managing a very high number of dimensions, imbalanced distribution and sparse attributes [79].

5) PROBLEM OF BIG DATA ANALYTICS IN TERMS OF DATA QUALITY

With the advent of big data, most data comes with new features, which may have serious data quality problems [105]. Big data is perceived to contain some undesirable contents that literally makes it non-useful. This is due to it is large volume and variety in terms of sources and formats with unstructured and semi-structured schema. These characteristics are the reasons why there is presence of noise, superfluous and/or inconsistency, missing of values [106], high dimension, and imbalance [105]. These problems get into big data due to different sources of it is acquisition and collation in a client to server architecture. The numerous client queries are directed to the collating server causing impediment on the server and hence distortion comes to play. The difference in the processing quality of the client devices and that of the communicating channel are the basis. These disparities resulted in noise in data from low quality device, poor and interference prone communication channels. Inconsistency and superfluous of data may be due to the difference in dimension in the generation of unstructured data. This variation leads to high dimensionality of such data. Also, missing of values may be resulted from human error in addition to the earlier problems [106]. Big data quality in terms of screening and simplification is important during preprocessing stage [105]. Quality results are gotten from quality data. Similarly, poor quality data result to poor results regardless of the intelligence of the algorithms applied to the task [107]. In machine learning, data preprocessing is highly required before the application of the ANN. The main purpose of the data preprocessing in machine learning is to generate high quality data. The data preprocessing phase involves data cleaning analysis, dimension reduction, subset selection of dimension, normalization, the correlation coefficient analysis of the dimensions, recovering missing values, resolving of discrepancies and input subset selection. Dealing with such data preprocessing issues at the level of big data analytics requires new form of processing technology to enable effective and efficient application of the ANN. Thereby, yield high quality results, hence, enhanced quality decision making.

VII. FUTURE RESEARCH OPPORTUNITIES

Despite the fact that the applications of ANN have given rise to improvements in big data analytics (see Table 5), many challenges are yet to be addressed. As such, future opportunities in the area of ANN applications in big data research can be summarised as follows:

In future applications of ANN in big data, the optimisation of the ANN should be performed using nature inspired algorithms to speed up the training process and enhance accuracy. The nature inspired algorithms are robust, and have the advantage of adaptation within a constantly changing setting [108]. These characteristics are likely to mean that the nature inspired algorithms perform well in terms of the exploration of big data. This is because the content of the big data is rapidly changing, and the nature inspired algorithm may cope with these constant changes since operations can be done automatically without requiring human intervention or effort. The nature inspired algorithms are likely to succeed in the exploration of big data in a real world environment due to their robustness. For an algorithm to be applied in a real world setting, it is required to be robust and to offer accuracy and convergence speed, as argued by Yang and Deb [109]. The optimization of the ANN using nature inspired algorithms can be done in different ways depending on the objective. For example, the determination of the optimal weight and biases of the ANN, and the automatic design of the ANN architectural structure and internal parameters without laborious trial and error efforts from the designer. The adaptation of the ANN learning rules can also be conducted using nature inspired algorithms.

We recommend the exploration of deep RNN in big data analytics since it has recorded an excellent performance on TIMIT phoneme recognition benchmark, and is considered as the best so far, as argued by Graves *et al*. [96]. The nature inspired algorithms described in Section VI, Sub-Section B (2) are recommended to researchers for training the ANN rather than using the commonly used backpropagation algorithm. The literature has shown that computational cost and energy consumption are a source of concern. As a result, the performance metrics of the applications of ANN in big data analytics should include communication costs and energy consumption, in addition to accuracy and convergence speed. This is likely to encourage researchers to propose a novel way of improving computational cost and energy efficiency in big data analytics. This is because the efforts are likely to be geared towards proposing an energy efficient ANN for application in big data analytics.

Reference [110] though, there is a school of thought that are critical about the training of the ANN using nature-inspired algorithms arguing that local minima is not a serious issue that warrant the use of nature-inspired algorithms for optimizing ANN. Overly, optimization of the ANN limit the ANN flexibility and lead to overfitting. It is highly scarce to find DNN train based on nature inspired algorithm except the ConvNet that was trained using simulated annealing and was found to improve the ConvNet performance. However, the second school of thought propagate the use of nature-inspired algorithms to train the ANN to improve its efficiency, robustness and efficiency as proven in the literature. Reference [111] propose hierarchical spatial-temporal state machine comprising of deep belief network, agglomerative hierarchical clustering, auto-encoders, and temporal sequence process for processing big data. We recommend researchers to extensively explore the framework in processing big data using DNN models. Reference [112] present an excellent tutorial on the implementation of ANN based on back-tracking on the CPU and GPU for big data analytics as well as the training of ANN across parallel machines. The framework presented in the tutorial has the potential for solving other classification, prediction, etc big data analytics problems based on DNN. Interested readers can read more on the tutorial in [112].

The managing of large data sets for inputs and outputs is difficult, and in the future, files should be managed as a subset in order to deviate from the challenges of managing inputs and outputs of a large dataset [113]. Smaller datasets, commonly used for evaluating ANN, should be relegated to the background and big data be put forward. Henceforth, ANN should be proposed with the capability to handle big data analytics since it is the modern day reality. The works in [114] and [115] can be extended to the ANN for big data analytics to explore it is potential for dealing with big data.

VIII. CONCLUSION

In this paper, we have presents a survey on the application of ANN approaches within the context of big data analytics. In our review, state-of-the-art issues regarding the application of ANN in big data analytics is provided. The several attempts made by researchers in the application of ANN are discussed. The paper has discussed the challenges involving ANN in terms of handling big data, and future research opportunities are unveiled. The progress within big data analytics using ANN is described. Research on the applications of ANN in big data analytics is at an early stage and is expected to grow rapidly in the near future. We believe that expert researchers can use this review as a benchmark for future progress and development. In addition, the paper can be used as a starting point for novice researchers interested in big data analytics using ANN.

CONFLICT OF INTEREST

The authors have no any conflict of interest reported.

REFERENCES

[1] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, "Data mining with big data," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 1, pp. 97–107, Jan. 2014.

[2] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani, and S. U. Khan, "The rise of 'big data' on cloud computing," *Inf. Syst.*, vol. 47, pp. 98–115, Jan. 2015.

[3] Y. Liu, J. Yang, Y. Huang, L. Xu, S. Li, and M. Qi, "MapReduce based parallel neural networks in enabling large scale machine learning," *Comput. Intell. Neurosci.*, Aug. 2015, Art. no. 297672.

[4] A. Hussain, E. Cambria, B. W. Schuller, and N. Howard, "Affective neural networks and cognitive learning systems for big data analysis," *Neural Netw.*, vol. 58, pp. 1–3, Oct. 2014.

[5] J. Abawajy, "Comprehensive analysis of big data variety landscape," *Int. J. Parallel, Emergent Distrib. Syst.*, vol. 30, no. 1, pp. 5–14, 2015.

[6] M. A. Kaboudan, "Compumetric forecasting of crude oil prices," in *Proc. Congr. Evol. Comput.*, May 2001, pp. 283–287.

[7] L. Yu, S. Wang, and K. K. Lai, "A generalized Intelligent-agent-based fuzzy group forecasting model for oil price prediction," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2008, pp. 489–493.

[8] A. Verikas, Z. Kalsyte, M. Bacauskiene, and A. Gelzinis, "Hybrid and ensemble-based soft computing techniques in bankruptcy prediction: A survey," *Soft Comput.*, vol. 14, no. 9, pp. 995–1010, 2010.

[9] D. Wan, Y. Xiao, P. Zhang, and H. Leung, "Hydrological big data prediction based on similarity search and improved BP neural network," in *Proc. IEEE Int. Congr. Big Data*, Jun./Jul. 2015, pp. 343–350.

[10] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "A survey on deep learning for big data," *Inf. Fusion*, vol. 42, pp. 146–157, Jul. 2018.

[11] Y. Wang, B. Li, R. Luo, Y. Chen, N. Xu, and H. Yang, "Energy efficient neural networks for big data analytics," in *Proc. Conf. Design, Autom. Test Eur.*, Mar. 2014, pp. 1–2.

[12] X. Yu and O. Kaynak, "Sliding-mode control with soft computing: A survey," *IEEE Trans. Ind. Electron.*, vol. 56, no. 9, pp. 3275–3285, Sep. 2009.

[13] H. Jaeger, "Adaptive nonlinear system identification with echo state networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 609–616.

[14] S. Scardapane, D. Wang, and M. Panella, "A decentralized training algorithm for echo state networks in distributed big data applications," *Neural Netw.*, vol. 78, pp. 65–74, Jun. 2015.

[15] J. Herbert, "Echo state network," *Scholarpedia*, vol. 2, no. 9, p. 2330, 2007.

[16] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[18] Y. LeCun *et al.*, "Handwritten digit recognition with a back-propagation network," in *Proc. Adv. Neural Inf. Process. Syst.*, 1990, pp. 396–404.

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[20] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*, vol. 1. Cambridge, MA, USA: MIT Press, 2016.

[21] M. Klassen, Y. H. Pao, and V. Chen, "Characteristics of the functional link net: A higher order delta rule net," in *Proc. IEEE ICNN*, Jun. 1988, pp. 507–513.

[22] J. C. Patra and R. N. Pal, "A functional link artificial neural network for adaptive channel equalization," *Signal Process.*, vol. 43, pp. 181–195, May 1995.

[23] Y. Yu, Y. Tian, N. Feng, and M. Lei, "Research on lifetime prediction method of tower crane based on back propagation neural network," in *Advances in Electronic Commerce, Web Application and Communication*. Berlin, Germany: Springer, 2012, pp. 111–116.

[24] M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Comput. Sci. Rev.*, vol. 3, no. 3, pp. 127–149, 2009.

[25] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[26] Y. Jin, "A comprehensive survey of fitness approximation in evolutionary computation," *Soft Comput.*, vol. 9, pp. 3–12, Jan. 2005.

[27] C. Wu, R. Buyya, and K. Ramamohanarao. (2016). "Big data analytics=machine learning+cloud computing." [Online]. Available: https://arxiv.org/abs/1601.03115

[28] M. Chen, S. Mao, and Y. Liu, "Big data: A survey," *Mobile Netw. Appl.*, vol. 19, no. 2, pp. 171–209, Apr. 2014.

[29] J. Manyika, J. Bughin, M. Chui, R. Dobbs, B. Brown, and C. Roxburgh, *Big Data: The Next Frontier for Innovation, Competition, and Productivity*. New York, NY, USA: McKinsey Global Institute, 2011.

[30] C. Ji *et al.*, "Big data processing: Big challenges and opportunities," *J. Interconnection Netw.*, vol. 13, Sep. 2012, Art. no. 1250009.

[31] K. Kambatla, G. Kollias, V. Kumar, and A. Grama, "Trends in big data analytics," *J. Parallel Distrib. Comput.*, vol. 74, no. 7, pp. 2561–2573, 2014.

[32] *CERN Computing*, CERN, Geneva, Switzerland, 2013.

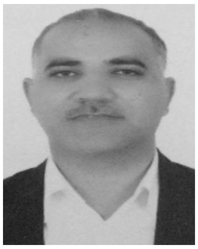
[33] Y. Chen, S. Alspaugh, and R. Katz, "Interactive analytical processing in big data systems: A cross-industry study of mapreduce workloads," *VLDB Endowment*, vol. 5, no. 12, pp. 1802–1813, 2012.
[34] Apache-Wiki. (2015). *Applications and Organizationsusing Hadoop*. [Online]. Available: http://wiki.apache.org/Hadoop/PoweredBy
[35] X. Zhao, H. Ma, H. Zhang, Y. Tang, and Y. Kou, "HVPI: Extending hadoop to support video analytic applications," in *Proc. IEEE 8th Int. Conf. Cloud Comput. (CLOUD)*, Jun./Jul. 2015, pp. 789–796.
[36] K. E. Taylor, R. J. Stouffer, and G. A. Meehl, "An overview of CMIP5 and the experiment design," *Bull. Amer. Meteorol. Soc.*, vol. 93, no. 4, pp. 485–498, 2012.
[37] R. Sumbaly, J. Kreps, and S. Shah, "The big data ecosystem at linkedin," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2013, pp. 1125–1134.
[38] A. Rossi, "80 terabytes of archived Web crawl data available for research," *Internet Arch. Blogs*, vol. 26, Oct. 2012. [Online]. Available: http://blog.archive.org/
[39] L. Clarke *et al.*, "The 1000 genomes project: Data management and community access," *Nature Methods*, vol. 9, pp. 459–462, Apr. 2012.
[40] G. Guzun, J. E. Tosado, and G. Canahuate, "Scalable preference queries for high-dimensional data using map-reduce," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Oct./Nov. 2015, pp. 2243–2252.
[41] C. Reiss, A. Tumanov, G. R. Ganger, R. H. Katz, and M. A. Kozuch, "Heterogeneity and dynamicity of clouds at scale: Google trace analysis," in *Proc. 3rd ACM Symp. Cloud Comput.*, 2012, Art. no. 7.
[42] S. Evert, "Google Web 1T 5-grams made easy (but not for the computer)," in *Proc. NAACL HLT 6th Web Corpus Workshop*, 2010, pp. 32–40.
[43] *Business Dictionary*. Accessed: Aug. 18, 2018. [Online]. Available: http://businessdictionary.org
[44] L. Wang *et al.*, "BigDataBench: A big data benchmark suite from Internet services," in *Proc. IEEE 20th Int. Symp. High Perform. Comput. Archit. (HPCA)*, Feb. 2014, pp. 488–499.
[45] A. Ghazal *et al.*, "BigBench: Towards an industry standard benchmark for big data analytics," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2013, pp. 1197–1208.
[46] J. Gray. (2012). *Graysort Benchmark*. [Online]. Available: http://sortbenchmark.org
[47] *GridMix-Benchmark*. Accessed: Jun. 16, 2017. [Online]. Available: http://hadoop.apache.org/docs/mapreduce/current/gridmix.html
[48] T. G. Armstrong, V. Ponnekanti, D. Borthakur, and M. Callaghan, "LinkBench: A database benchmark based on the Facebook social graph," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2013, pp. 1185–1196.
[49] *PigMix-Benchmark*. Accessed: Jun. 16, 2017. [Online]. Available: https://cwiki.apache.org/confluence/display/PIG/PigMix
[50] M. Ferdman *et al.*, "Clearing the clouds: A study of emerging scale-out workloads on modern hardware," in *Proc. ACM SIGPLAN Notices*, 2012, pp. 37–48.
[51] B. F. Cooper, A. Silberstein, E. Tam, R. Ramakrishnan, and R. Sears, "Benchmarking cloud serving systems with YCSB," in *Proc. 1st ACM Symp. Cloud Comput.*, 2010, pp. 143–154.
[52] AMPLab. *Big Data Benchmark*. [Online]. Available: https://amplab.cs.berkeley.edu/benchmark/
[53] AMPLab. *Big Data Benchmark*. Accessed: Aug. 18, 2018. [Online]. Available: https://amplab.cs.berkeley.edu/benchmark/
[54] S. Huang, J. Huang, J. Dai, T. Xie, and B. Huang, "The HiBench benchmark suite: Characterization of the MapReduce-based data analysis," in *Proc. IEEE 26th Int. Conf. Data Eng. Workshops (ICDEW)*, Mar. 2010, pp. 41–51.
[55] D. Karaboga and C. Ozturk, "A novel clustering approach: Artificial bee colony (ABC) algorithm," *Appl. Soft Comput.*, vol. 11, pp. 652–657, Jan. 2011.
[56] O. L. Mangasarian, W. N. Street, and W. H. Wolberg, "Breast cancer diagnosis and prognosis via linear programming," *Oper. Res.*, vol. 43, no. 4, pp. 570–577, Aug. 1995.
[57] I. Guyon, S. Gunn, A. Ben-Hur, and G. Dror, "Result analysis of the NIPS 2003 feature selection challenge," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 545–552.
[58] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.
[59] P. W. Frey and D. J. Slate, "Letter recognition using Holland-style adaptive classifiers," *Mach. Learn.*, vol. 6, no. 2, pp. 161–182, 1991.
[60] R. P. Gorman and T. J. Sejnowski, "Analysis of hidden units in a layered network trained to classify sonar targets," *Neural Netw.*, vol. 1, no. 1, pp. 75–89, 1988.
[61] R. J. Alcock and Y. Manolopoulos, "Time-series similarity queries employing a feature-based approach," in *Proc. 7th Hellenic Conf. Inform.*, 1999, pp. 27–29.
[62] S. Hettich and S. Bay, "The UCI KDD archive," Dept. Inf. Comput. Sci., Univ. California, Irvine, Irvine, CA, USA, Tech. Rep. 152, 1999, vol. 152. [Online]. Available: http://kdd.ics.uci.edu
[63] H. A. Güvenir, G. Demiröz, and N. Ilter, "Learning differential diagnosis of erythemato-squamous diseases using voting feature intervals," *Artif. Intell. Med.*, vol. 13, pp. 147–165, Jul. 1998.
[64] J. R. Quinlan, P. J. Compton, K. A. Horn, and L. Lazarus, "Inductive knowledge acquisition: A case study," in *Proc. 2nd Austral. Conf. Appl. Expert Syst.*, 1987, pp. 137–156.
[65] R. Kohavi, "Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid," in *Proc. KDD*, 1996, pp. 202–207.
[66] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomputing*, vol. 171, pp. 754–767, Jan. 2016.
[67] A. Stisen *et al.*, "Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition," in *Proc. 13th ACM Conf. Embedded Networked Sensor Syst.*, 2015, pp. 127–140.
[68] O. Y. Al-Jarrah, P. D. Yoo, S. Muhaidat, G. K. Karagiannidis, and K. Taha, "Efficient machine learning for big data: A review," *Big Data Res.*, vol. 2, no. 3, pp. 87–93, 2015.
[69] A. Stateczny and M. Wlodarczyk-Sielicka, "Self-organizing artificial neural networks into hydrographic big data reduction process," in *Proc. Int. Conf. Rough Sets Intell. Syst. Paradigms*, 2014, pp. 335–342.
[70] X. Chen, H. H. Wang, and B. Tian, "Visualization model of big data based on self-organizing feature map neural network and graphic theory for smart cities," *Cluster Comput.*, pp. 1–13, Jan. 2018.
[71] A. Rosà, L. Y. Chen, and W. Binder, "Catching failures of failures at big-data clusters: A two-level neural network approach," in *Proc. IEEE 23rd Int. Symp. Qual. Service (IWQoS)*, Jun. 2015, pp. 231–236.
[72] A. P. Tafti *et al.*, "bigNN: An open-source big data toolkit focused on biomedical sentence classification," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 3888–3896.
[73] S. YiLiang and Y. Zhenghong, "Communication rules learning strategy in big data network based on SVN neural network," in *Proc. Int. Conf. Intell. Transp., Big Data Smart City (ICITBS)*, Dec. 2016, pp. 309–313.
[74] V. P. Barros and P. Notargiacomo, "Big data analytics in cloud gaming: Players' patterns recognition using artificial neural networks," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2016, pp. 1680–1689.
[75] S. Fong, C. Fang, N. Tian, R. Wong, and B. W. Yap, "Self-adaptive parameters optimization for incremental classification in big data using neural network," in *Big Data Applications and Use Cases*. Cham, Switzerland: Springer, 2016, pp. 175–196.
[76] Y. Liu, L. Xu, and M. Li, "The parallelization of back propagation neural network in MapReduce and Spark," *Int. J. Parallel Program.*, vol. 45, pp. 760–779, Aug. 2017.
[77] H. Shen *et al.*, "A quantitative quality control method of big data in cancer patients using artificial neural network," in *Proc. IEEE 3rd Int. Conf. Cloud Comput. Intell. Syst. (CCIS)*, Nov. 2014, pp. 499–504.
[78] J. N. Liu, Y. Hu, Y. He, P. W. Chan, and L. Lai, "Deep neural network modeling for big data weather forecasting," in *Information Granularity, Big Data, and Computational Intelligence*. Cham, Switzerland: Springer, 2015, pp. 389–408.
[79] E. Elsebakhi *et al.*, "Large-scale machine learning based on functional networks for biomedical big data with high performance computing platforms," *J. Comput. Sci.*, vol. 11, pp. 69–81, Nov. 2015.
[80] I.-H. Chung *et al.*, "Parallel deep neural network training for big data on blue gene/q," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 6, pp. 1703–1714, Jun. 2017.
[81] R. Gu, F. Shen, and Y. Huang, "A parallel computing platform for training large scale neural networks," in *Proc. IEEE Int. Conf. Big Data*, Oct. 2013, pp. 376–384.
[82] W. Aziguli *et al.*, "A robust text classifier based on denoising deep neural network in the analysis of big data," *Sci. Program.*, Nov. 2017, Art. no. 3610378.
[83] J. Cao, H. Cui, H. Shi, and L. Jiao, "Big data: A parallel particle swarm optimization-back-propagation neural network algorithm based on MapReduce," *PloS ONE*, vol. 11, no. 6, 2016, Art. no. e0157551.

[84] D. Chen, "Research on traffic flow prediction in the big data environment based on the improved RBF neural network," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 2000–2008, Aug. 2017.
[85] H. Hu, B. Tang, X. Gong, W. Wei, and H. Wang, "Intelligent fault diagnosis of the high-speed train with big data based on deep neural networks," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 2106–2116, Aug. 2017.
[86] S. J. Plathottam, H. Salehfar, and P. Ranganathan, "Convolutional neural networks (CNNs) for power system big data analysis," in *Proc. North Amer. Power Symp. (NAPS)*, Sep. 2017, pp. 1–6.
[87] N. Straton, R. R. Mukkamala, and R. Vatrapu, "Big social data analytics for public health: Predicting Facebook post performance using artificial neural networks and deep learning," in *Proc. IEEE Int. Congr. Big Data (BigData Congr.)*, Jun. 2017, pp. 89–96.
[88] Z. Zhang, D. Odaibo, F. M. Skidmore, and M. M. Tanik, "A big data analytics approach in medical imaging segmentation using deep convolutional neural networks," in *Big Data and Visual Analytics*. Cham, Switzerland: Springer, 2017, pp. 181–189.
[89] R. Zhang, W. Deng, and M. Y. Zhu. (2017). "Using deep neural networks to automate large scale statistical analysis for big data applications." [Online]. Available: https://arxiv.org/abs/1708.03027
[90] I. Safder, S.-Ul Hassan, and N. R. Aljohani, "AI cognition in searching for relevant knowledge from scholarly big data, using a multi-layer perceptron and recurrent convolutional neural network model," in *Proc. Web Conf. Companion Web Conf.*, 2018, pp. 251–258.
[91] J. Amin, M. Sharif, M. Yasmin, and S. L. Fernandes, "Big data analysis for brain tumor detection: Deep convolutional neural networks," *Future Gener. Comput. Syst.*, vol. 87, pp. 290–297, Oct. 2018.
[92] L. Oneto *et al.*, "Train delay prediction systems: A big data analytics perspective," *Big Data Res.*, vol. 11, pp. 54–64, Mar. 2018.
[93] D. Triantafyllidou, P. Nousi, and A. Tefas, "Fast deep convolutional face detection in the wild exploiting hard sample mining," *Big Data Res.*, vol. 11, pp. 65–76, Mar. 2018.
[94] R. Zhang, "Research on big data feature analysis based on kernel discriminant analysis and neural network," *Multimedia Tools Appl.*, Feb. 2018, pp. 1–13.
[95] S.-W. Park *et al.*, "An energy-efficient and scalable deep learning/inference processor with tetra-parallel MIMD architecture for big data applications," *IEEE Trans. Biomed. Circuits Syst.*, vol. 9, no. 6, pp. 838–848, Dec. 2015.
[96] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2013, pp. 6645–6649.
[97] E. Rueckert, D. Kappel, D. Tanneberg, D. Pecevski, and J. Peters, "Recurrent spiking networks solve planning tasks," *Sci. Rep.*, vol. 6, Feb. 2016, Art. no. 21142.
[98] A. Ponsich, A. L. Jaimes, and C. A. C. Coello, "A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications," *IEEE Trans. Evol. Comput.*, vol. 17, no. 3, pp. 321–344, Jun. 2013.
[99] D. Karaboga, B. Akay, and C. Ozturk, "Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks," in *Proc. Int. Conf. Modeling Decis. Artif. Intell.*, 2007, pp. 318–329.
[100] N. M. Nawi, A. Khan, M. Z. Rehman, H. Chiroma, and T. Herawan, "Weight optimization in recurrent neural networks with hybrid metaheuristic Cuckoo search techniques for data classification," *Math. Problems Eng.*, vol. 2015, Nov. 2015, Art. no. 868375.
[101] H. Chiroma *et al.*, "Global warming: Predicting OPEC carbon dioxide emissions from petroleum consumption using neural network and hybrid cuckoo search algorithm," *PloS ONE*, vol. 10, no. 8, 2015, Art. no. e0136140.
[102] B. Xing and W.-J. Gao, *Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms*. Melbourne, VIC, Australia: Springer, 2014.
[103] I. Fister, Jr, X.-S. Yang, I. Fister, J. Brest, and D. Fister. (2013). "A brief review of nature-inspired algorithms for optimization." [Online]. Available: https://arxiv.org/abs/1307.4186
[104] C. Pettey. (2013). Gartner estimates ICT industry accounts for 2 percent of global CO_2 emissions. DOSTUPNO. [Online]. Available: https://www.gartner.com/newsroom/id
[105] Z.-H. Zhou, N.-V. Chawla, Y. Jin, and G. J. Williams, "Big data opportunities and challenges: Discussions from data analytics perspectives [discussion forum]," *IEEE Comput. Intell. Mag.*, vol. 9, no. 4, pp. 62–74, Nov. 2014.
[106] A. Oussous, F. Z. Benjelloun, A. A. Lahcen, and S. Belfkih, "Big Data technologies: A survey," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 30, no. 4, pp. 431–448, 2018.
[107] S. Zhang, C. Zhang, and Q. Yang, "Data preparation for data mining," *Appl. Artif. Intell.*, vol. 17, nos. 5–6, pp. 375–381, 2003.
[108] H. A. Abbass, "A memetic Pareto evolutionary approach to artificial neural networks," in *Proc. Austral. Joint Conf. Artif. Intell.*, 2001, pp. 1–12.
[109] X.-S. Yang and S. Deb, "Cuckoo search: Recent advances and applications," *Neural Comput. Appl.*, vol. 24, no. 1, pp. 169–174, Jan. 2014.
[110] S. Fong, S. Deb, and X.-S. Yang, "How meta-heuristic algorithms contribute to deep learning in the hype of big data analytics," in *Progress in Intelligent Computing Techniques: Theory, Practice, and Applications*. Singapore: Springer, 2018, pp. 3–25.
[111] R. Iqbal, F. Doctor, B. More, S. Mahmud, and U. Yousuf, "Big data analytics and computational intelligence for cyber–physical systems: recent trends and state of the art applications," *Future Gener. Comput. Syst.*, to be published. doi: 10.1016/j.future.2017.10.021.
[112] A. Fonseca and B. Cabral, "Prototyping a GPGPU neural network for deep-learning big data analysis," *Big Data Res.*, vol. 8, pp. 50–56, Jul. 2017.
[113] B. C. Pijanowski, A. Tayyebi, J. Doucette, B. K. Pekin, D. Braun, and J. Plourde, "A big data urban growth simulation at a national scale: Configuring the GIS and neural network based land transformation model to run in a high performance computing (HPC) environment," *Environ. Model. Softw.*, vol. 51, pp. 250–268, Jan. 2014.
[114] W. Gong, Y. Wang, Z. Cai, and S. Yang, "A weighted biobjective transformation technique for locating multiple optimal solutions of nonlinear equation systems," *IEEE Trans. Evol. Comput.*, vol. 21, no. 5, pp. 697–713, Oct. 2017.
[115] W. Gong, A. Zhou, and Z. Cai, "A multioperator search strategy based on cheap surrogate models for evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 746–758, Oct. 2015.

HARUNA CHIROMA received the B.Tech. degree in computer science from Abubakar Tafawa Balewa University, Bauchi, Nigeria, and the M.Sc. degree in computer science from Bayero University Kano, Nigeria, and the Ph.D. degree from the Department of Artificial Intelligence, Faculty of Computer Science and Information Technology, University of Malaya. He is currently a Senior Lecturer with the Federal College of Education (Technical), Gombe, Nigeria. He has published over 100 articles relevant to his research interest in international referred journals, edited books, conference proceedings and local journals. His main research interest includes meta-heuristic algorithms, soft computing, big data, data mining, machine learning, human–computer interaction, software engineering, and information security and recently, social media in education. He is a member of the ACM. He served in the technical program committee of several international conferences.

USMAN ALI ABDULLAHI received the B.Tech. degree (Hons.) in computer science from Abubakar Tafawa Balewa University, Bauch, Nigeria, in 2003, and the M.Sc. degree in computer science from Bayero University Kano in 2011. He is currently pursuing the Ph.D. degree with Universiti Teknologi PETRONAS, Malaysia. He started working as a Lecturer with the Federal College of Education (Tech) Gombe in 2004. His areas of research interest are software engineering, database programming, big data, MapReduce, and advanced database.

SHAFI'I MUHAMMAD ABDULHAMID received the B.Tech. degree in mathematics/computer science from the Federal University of Technology Minna, Nigeria, and the M.Sc. degree in computer science from Bayero University Kano, Nigeria. He is currently pursuing the Ph.D. degree with the Faculty of Computing, University of Technology, Malaysia. He is currently a Lecturer with the Federal University of Technology Minna. His current research interests are in grid computing, cloud computing, big data, and soft computing. He is a member of the Computer Professional Registration Council of Nigeria, the Computer Society, and the Nigerian Computer Society (NCS).



ALA ABDUSALAM ALAROOD received the bachelor's and master's degrees from Yarmok University, Jordan, and the Ph.D. degree from the University of Technology, Malaysia, all in computer science. He is currently an Assistant Professor of computer science with the Faculty of Computer and Information Technology, University of Jeddah. His current research interest is in the area of information security, networks security, steganalysis, machine learning, and neural network.

LUBNA A. GABRALLA received the M.Sc. and Ph.D. degrees in computer science from the Sudan University of Science and Technology, Khartoum, Sudan. She is currently an Assistant Professor with the Department of Computer Science and Information Technology, Princess Nourah Bint Abdulrahman University, Saudi Arabia. Her current research interests are in artificial neural network, machine learning, and cyber security.

NADIM RANA received the M.Sc. degree in computer science from the University of Technology Malaysia, where he is currently pursuing the Ph.D. in computer science. His current research interests include cloud computing, nature inspired algorithms, machine learning, and cyber security.

LIYANA SHUIB received the B.Sc. degree (Hons) in computer science from the University of Technology, Malaysia, Skudai, Malaysia, the master's degree in information technology from the Universiti Kebangsaan Malaysia, and the Ph.D. degree from the University of Malaya. She is a currently a Senior Lecturer with the Department of Information System, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur. She graduated Ph.D. student and currently supervising several post graduate students. She has more than 20 papers relevant to her research interest.

IBRAHIM ABAKER TARGIO HASHEM received the B.E. degree in computer science Sudan in 2007 and the M.S. degree in computing Malaysia in 2012. He is currently pursuing the Ph.D. degree with the Department of Computer Systems and Technology, University of Malaya, Kuala Lumpur, Malaysia. He received professional certificates from CISCO (CCNP, CCNA, and CCNA Security) and APMG Group (PRINCE2 Foundation, ITIL v3 Foundation, and OBASHI Foundation). He was a Tutor with the CISCO Academy, University of Malaya. His main research interests include big data, cloud computing, distributed computing, and network.

DADA EMMANUEL GBENGA He is currently a Senior Lecturer with the University of Maiduguri. received the M.Sc. degree from University of Maiduguri and the Ph.D. degree from the Department of Artificial Intelligence, University of Malaya, Kuala Lumpur, Malaysia. His main research interests include nature inspired algorithms, robotics, big data, and machine learning.

ADAMU I. ABUBAKAR received the B.Sc. degree in geography, P.G.D., M.Sc., and Ph.D. degrees in computer science. He is currently an Assistant Professor with the International Islamic University of Malaya, Kuala Lumpur, Malaysia. He has published over 100 articles relevant to his research interest.

AKRAM M. ZEKI received the Ph.D. degree in digital watermarking from University of Technology Malaysia. He is currently an Associated Professor from the Information System Department, International Islamic University Malaysia. His current research interest is on protection of watermarking.

TUTUT HERAWAN received the Ph.D. degree in information technology from Universiti Tun Hussein Onn Malaysia in 2010. He is currently an Associate Professor with the Department of Information Systems, Faculty of Computer Science and Information Technology, University of Malaya. He has successfully supervised many Ph.D. students and currently supervised more than 10 Ph.D. students. He has edited many Springer book series and published many articles in various international journals, conference proceedings, and book chapters. His research area includes rough and soft set theory, soft computing for data mining, and knowledge discovery in databases. He has served as the chair, a PC chair, a program committee member, and co-organizer for numerous international conferences/workshops. He is the founding chair of two international conferences DaEng and SCDM. He has been appointed as an Associate Editor for *Malaysian Journal of Computer Science* (*SCIE*) and *TELKOMNIKA Journal* (*Scopus*).