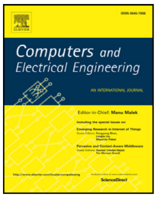
Computers and Electrical Engineering 000 (2017) 1–13

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng



Deep learning in big data Analytics: A comparative study[☆]

Bilal Jan[a], Haleem Farman[b], Murad Khan[c], Muhammad Imran[c], Ihtesham Ul Islam[c], Awais Ahmad[d,*], Shaukat Ali[b], Gwanggil Jeon[e,*]

[a] Department of Computer Science, FATA University, FR Kohat, Pakistan
[b] Department of Computer Science, Islamia College Peshawar, Pakistan
[c] Department of Computer Science and IT, Sarhad University of Science & IT, Peshawar, Pakistan
[d] Department of Information and Communication Engineering, Yeungnam University, Gyeongsan, South Korea
[e] Department of Embedded Systems Engineering, College of Information and Technology, Incheon National University, South Korea

ARTICLE INFO

Article history:
Received 2 June 2017
Revised 25 November 2017
Accepted 2 December 2017
Available online xxx

Keywords:
Big data
Deep learning
Deep belief networks
Convolutional Neural Networks

ABSTRACT

Deep learning methods are extensively applied to various fields of science and engineering such as speech recognition, image classifications, and learning methods in language processing. Similarly, traditional data processing techniques have several limitations of processing large amount of data. In addition, Big Data analytics requires new and sophisticated algorithms based on machine and deep learning techniques to process data in real-time with high accuracy and efficiency. However, recently, research incorporated various deep learning techniques with hybrid learning and training mechanisms of processing data with high speed. Most of these techniques are specific to scenarios and based on vector space thus, shows poor performance in generic scenarios and learning features in big data. In addition, one of the reason of such failure is high involvement of humans to design sophisticated and optimized algorithms based on machine and deep learning techniques. In this article, we bring forward an approach of comparing various deep learning techniques for processing huge amount of data with different number of neurons and hidden layers. The comparative study shows that deep learning techniques can be built by introducing a number of methods in combination with supervised and unsupervised training techniques.

1. Introduction

In this fast-growing digital world, Big Data and Deep learning are the high attention of data science. Big Data is the collection of huge amount of digital raw data that is difficult to manage and analyse using traditional tools [1,2]. As the digital data is growing exponentially in different shapes, formats and sizes, therefore it is very important to manage this large volume of data according to the needs of the organization. The companies based on technology such as Microsoft, Yahoo, Amazon and Google have maintained data in Exabyte or even larger. Due to the popularity of social online media companies such as YouTube, Twitter and Facebook, a huge amount of data is generated by billions of users. However, this bulk of information cannot be managed by conventional tools. Therefore, different organizations have developed products by using Big Data Analytics for experimentation, simulations, data analysis, monitoring and many more business needs,

[☆] Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. A. K. Sangaiah.

[*] Corresponding authors.

E-mail addresses: bilal.jan@fu.edu.pk (B. Jan), haleem.farman@icp.edu.pk (H. Farman), murad.csit@suit.edu.pk (M. Khan), Imran.csit@suit.edu.pk (M. Imran), ihtesham.csit@suit.edu.pk (I.U. Islam), aahmad.marwat@gmail.com (A. Ahmad), shaukat@icp.edu.pk (S. Ali), gjeon@inu.ac.kr (G. Jeon).

https://doi.org/10.1016/j.compeleceng.2017.12.009
0045-7906/© 2017 Elsevier Ltd. All rights reserved.

which makes it an important topic of data science. The principal task of Big Data Analytics is to extract useful patterns from the huge amount of data that can be used in decision making and prediction. However, there are some other challenges that Big Data Analytics faces for data analysis and machine learning such as different formats and sizes of input data, fast data streaming, data analysis reliability, quality of data, un-categorized and un-supervised input data, quick retrieval of information, data tagging, and data storage, etc. [1,2].

Big Data has the capability to revolutionize almost all parts of the society, collecting and managing useful data from Big Data is quite difficult and complex task. The rapidly expanding body of hidden information in a huge bulk of nontraditional data needs some advanced technologies to be developed along with the multidisciplinary expert team. Machine learning techniques along with computational power have significant part in Big Data analytics. Machine learning focused on input data representation and learnt patterns generalization to be predicted for future data [3]. The representation of data has quite an effect on machine learner's performance. A good data representation can result in high performance, even if the machine learner is simple while the poor representation of data with advance complex machine learner might lead to decreased performance. Therefore, a key element of machine learning known as feature engineering is used to construct features and represent data from raw input data. A large effort is required for feature engineering and is usually domain specific. Machine learning is widely deployed to explore the predictive feature of Big Data in many fields such as medicine, Internet of Things (IoT), search engines and much more. To deal with Big Data analytics, an important sub-field of machine learning known as deep learning is used to extract useful data out of the Big Data [4].

In comparison with the conventional learning techniques, which considers shallow structured architectures which do not use in depth learning, deep learning uses supervised and unsupervised techniques using machine learning approaches to learn hierarchical data representation automatically for feature classification. Deep learning has inspiration from the human brain representation for natural signals processing; it attracted the academic community in the recent years due to its performance in different research areas such as medical, computer vision, speech recognition and much more. Moreover, technological companies Facebook, Apple and Google collect and analyse huge amount of digital data daily and are seriously taking a keen interest in projects related to deep learning. For instance, the iPhone (Apple's product) virtual personal assistant named as Siri, collects data from the customer and according to perform the tasks use deep learning. Moreover, it offers a variety of different tasks such as setting the alarm, news, weather reports, send payment, and even one can change the lighting of the room as well. The more you use this application, the more it gets to know what you need at any specific time. Google also takes advantages of deep learning algorithms for Google's translator, image and video searching and Android's voice recognition. Companies such as Microsoft and IBM are also taking advantages of deep learning techniques.

The main contribution of this paper is to compare different deep learning techniques in context of big data processing. Two main types of deep learning techniques is studies to show the importance of the deep learning techniques for processing large data. The rest of the paper is divided into following sections. Section 2 presents an explanation of big data analytics followed by deep learning in big data in Section 3. A comparative study of the deep learning techniques along with experimental results are given in Section 4. Finally, the conclusion is presented in Section 5.

2. Big data analytics

The world has transformed into digital world, where a huge amount of data is generated day-by-day from different platforms that gave rise to Big Data. It deals with huge volumes of raw data. Big Data is used in almost all walk of life such as business, public administration, national security, scientific research, healthcare, Internet of Things (IoT), commercial recommendations, stock exchanges and many more. Due to the popularity of online social network that enabled billions of customers to upload different types and sizes of data such as text, images and videos, that forced the technological and social media organizations like Google, Amazon, Facebook and Twitter etc. to think seriously regarding this bulk of data increasing day by day. The main challenge of such size of data is not just in collecting it but to manage it properly and to make use of it in efficient way for decision making or prediction. To deal with Big Data problems, several tools are designed and developed to handle this huge amount of dissimilar and complex data to get further benefits out of it [1]. Fig. 1, depicts big data literature classification by highlighting challenges, data types, big data tools and applications.

2.1. Challenges in big data analytics

Apart from this huge amount of data, Big Data has some other complications usually referred as four Vs; Volume, Variety, Velocity and Veracity [5]. The unstructured and un-labelled large Volume of data is one of the key feature of Big Data as well, as it can deal with this amount of data. As the generic concept of Big Data is to deal with dissimilar and complex raw input data, that mainly consist of unsupervised data of different sizes and might have a little portion of supervised data. Therefore, this Variety of data representation give rise to different challenges in Big Data to extract useful and structured data out of unstructured and un-categorized data. In today's era, the rate at which the data is produced and stored makes the data Velocity equally important as volume and variety. If the data is not processed on time, chances of data loss increases especially in streaming data. Therefore, it is important for Big Data tools to timely process the data and translate it in to effective information. Different organizations such as IBM, Facebook and Twitter have designed and developed products to deal with data streaming. Veracity mean accuracy, trustworthiness and validity of results attained from the analysis. Due to the increasingly different data sources and variety, accuracy and trust becomes a challenge in Big Data Analytics. Besides the

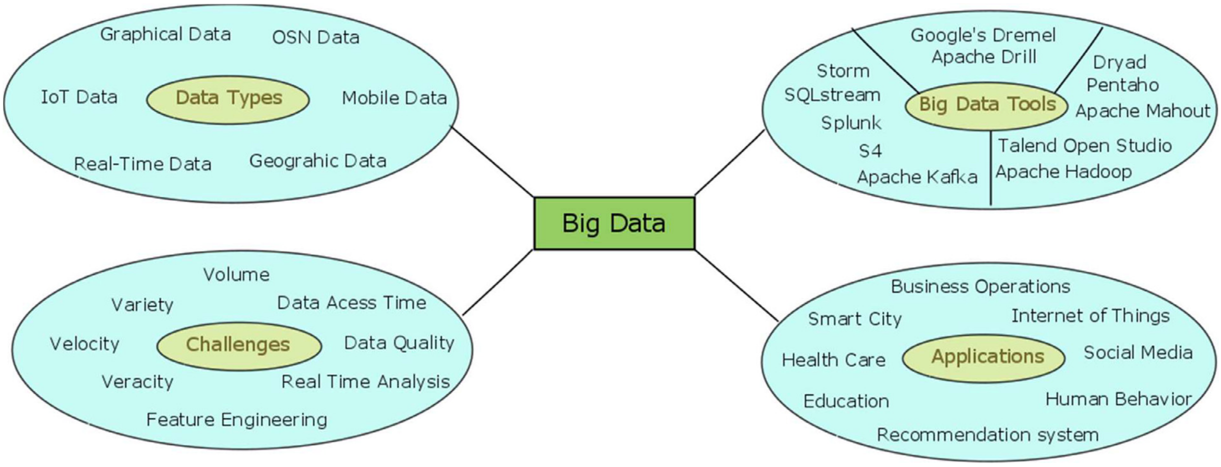


Fig. 1. Big Data literature classification.

four Vs, there are number of other challenges in Big Data Analytics, few of them are: feature engineering, data quality, data representation from various sources, parallel and distributed data processing, integration of different data, data discovery, data access time, real time analysis, sampling of data and computation of huge volume of data [6].

2.2. Data type

Number of different datasets are produced due to the emergence of Big Data, most of the data sets are domain specific having different data representation, density, distribution and sizes of data. To extract knowledge from these datasets is of high significance in Big Data research that makes it distinguish from conventional data mining [7]. Few of the data types are listed below that dealt with Big Data.

2.2.1. Online social network data

The online social network (OSN) data comes under network big data. A variety of data is uploaded on daily basis by millions of users using different online social media such as Facebook, Twitter etc. Due to the increasing popularity of these online social media, has put up some challenges for managing this huge volume of data which attracted the researchers to come up with the efficient data analysis tools. Many studies has been carried out using knowledge regarding characteristics representations at high-level and low-level abstraction. For Instance, Facebook uses some learning techniques to manage user's data and to recommend friends or pages in which the user might be interested.

2.2.2. Internet of things data

This new concept of Internet of Things, will generate huge amount of data as everything will be connected to internet having an address that can be controlled remotely. Large number of sensors attached to different devices will generate diverse and complex data that will be send to a central location for further processing or for any decision making. This bulk of data cannot be efficiently processed with the traditional data mining techniques, therefore Big Data tools are required to deal with such diverse un-supervised and un-labelled data [8,9].

2.2.3. Mobile data

The mobile communication is developed to great deal with the introduction of 4 G and 5 G technologies. The traditional cellular network is transformed in to converged mobile network and has brought substantial improvement in communications. These days huge volume of data is generated using mobile devices. As the use of mobile has substantially increased due to diverse applications offered by mobile developers such as use of google maps for easy location access and to find nearby options available for users.

2.2.4. Real-time data

The network big data has changed from spatial temporal data to real time spatial time-based data due to the increase in OSN data using streaming services. The use of social networks has changed the concept of data streaming to big data streaming. The social network's data streams and other networks relying on sensors has introduced a new approach to extended figures analysis. The data obtained from these figures is collected from servers at different locations. Depending on stream handling engine, authors in [10] proposed a framework named as Floe. It allows real-time data handling and extended figure renewal to analyze figures in large-scale with minimum delay in continuously varying OSN. Furthermore, it can be extended to different walk of life such as health supervision and anti-terrorist applications [10].

Please cite this article as: B. Jan et al., Deep learning in big data Analytics: A comparative study, Computers and Electrical Engineering (2017), https://doi.org/10.1016/j.compeleceng.2017.12.009

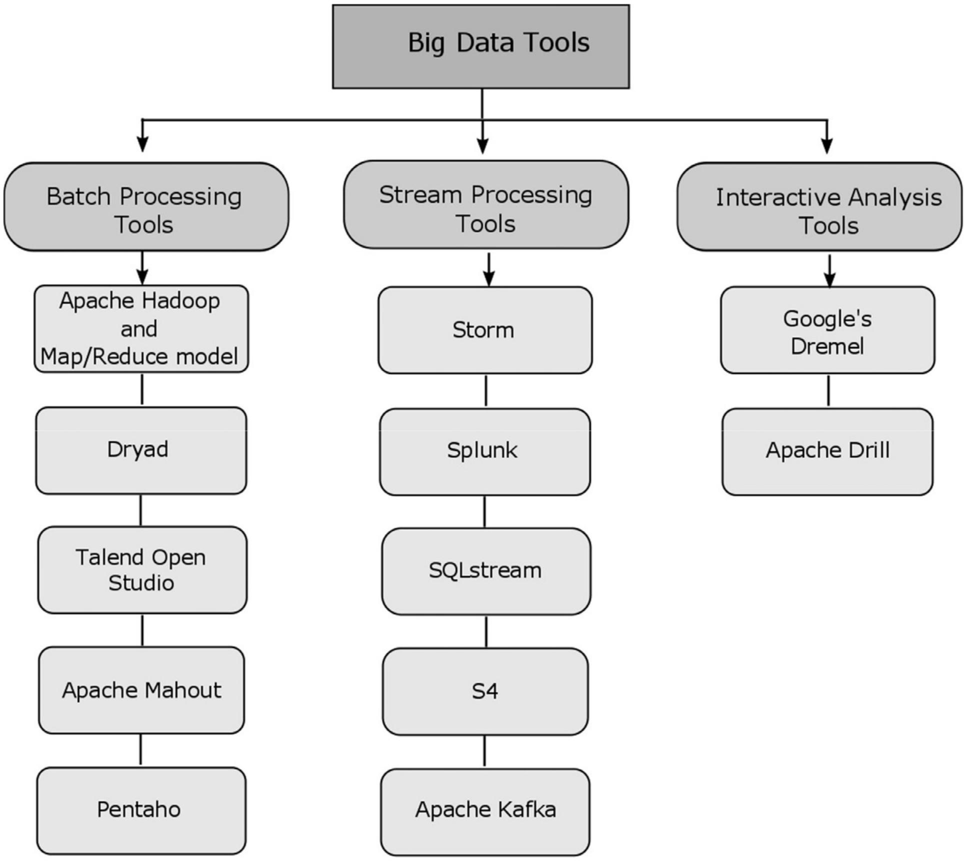


Fig. 2. Big Data Tools.

The data that changes rapidly is termed as real-time data such as the data collected and monitored of City traffic. In VANETs, the traffic data collected from vehicles is very important to guide humans and to update them regarding road status. In [11], a framework was proposed for real-time traffic data in VANETs, which is composed of focused data memory principles for number of processes and distributed memory principles for data streaming in real-time [11]. In industrial applications, most of the data is used for the analysis of risk management. In a complex indoor environment, there are some restrictions in collecting big data. To overcome these restrictions, indoor sensors are deployed to collect big data, which is then forwarded to a central data point for further analysis or decision making [12]. In [13], a technique based on indoor wireless sensor network was proposed for gathering big data to analyze the industrial risk management.

2.2.5. Graphical data

As earlier discussed, due to the increasingly use of OSN, the graphical data especially images has opted challenges in the field of image processing. Authors in [14] introduced weakly semi-supervised deep learning (WeSed) technique for multi-label image representation. To deal with the weakly-labelled images, they introduces a novel weakly weighted pairwise unlabeled classifications. WeSed is capable to train deep convolutional neural network (CNN). Images obtained from OSN are often weakly labelled, multi-labelled or some might have no label. In [15], robust discrete hashing (RDSH) is proposed, an un-supervised hashing technique for pictorial data using large scale semantic indexing. RDSH learn hash functions and discrete binary codes simultaneously in a single model. The proposed method in terms of large scale semantic indexing was compared with state of the art techniques using different image datasets.

2.3. Big data tools

To deal with Big Data, we need to have specialized tools that can handle the huge complex form of data in an efficient and effective way. As the traditional tools cannot manage the analytics of Big Data, therefore few of the tools available are discussed here. The Big data tools are mainly classified in to batch processing, stream processing and interactive analysis [16] as shown in Fig. 2, each of them is discussed below.

Please cite this article as: B. Jan et al., Deep learning in big data Analytics: A comparative study, Computers and Electrical Engineering (2017), https://doi.org/10.1016/j.compeleceng.2017.12.009

2.3.1. Batch processing tools

Apache Hadoop is the most dominant batch processing tool used in Big Data, as most of the applications used Hadoop as a platform. Hadoop is widely used in different domains such as machine learning and data mining etc. Hadoop distributes load through different machines. It works well in large data processing as it is specially designed for batch processing. Here some of the batch processing tools are briefly explained.

2.3.1.1. Apache Hadoop and Map/Reduce model. The Apache Hadoop is a software platform that is developed for distributed data-intensive applications. Apache Hadoop uses Map/Reduce as a computational paradigm [16]. Map/Reduce is a programming model established by Google and other web companies to process, analyze and generate huge data sets. It breaks down a complex problem in to sub-problems and this process continues until each sub-problem can be directly handled. Processing clusters are used to solve these sub-problems in parallel and at the end they are combined to give solid solution to the main problem. The Map/Reduce method works in two steps namely map and reduce. Moreover, the Hadoop architecture consists of two kinds of nodes: master and slave nodes. In Map step, the master nodes divides the whole problem in to sub-problems and then distributes it among slave nodes for further processing. The slave nodes processes it and then in reduce step, the solutions to the sub-problems are forwarded to master nodes to combine it for one final solution. Integrating Map/Reduce model in to Hadoop architecture, makes it more powerful framework for parallel processing of huge data sets on large clusters in more reliable way. The Map/Reduce framework is more explained in detail along with some other data analysis tools in [17].

2.3.1.2. Dryad. A programming model named as Dryad, which can process programs in parallel and distributed manner. It has the processing capability from small cluster (having few number of nodes) to very large cluster (consisting even thousands of nodes). It uses the concept of cluster (having computing nodes) to execute and process programs in a way distributed manner. The programmers using Dryad framework can work using hundreds and thousands of machines having multiple cores or processors. Moreover, one of the advantage using Dryad framework is that programmers doesn't need to know about the parallel programming. Applications using this framework runs on directed graph, consists of vertices and edges. Where programs are represented by vertices and edges represents the communicational channels. A series of programs are linked by one-way channels. In Dryad the vertices are connected with each other in acyclic manner and during computation if some unexpected events occurs, it can the capability to update the graphs even after execution. Moreover, Dryad offers functionalities such as graph generation, processes scheduling, handling errors in a cluster, handling user-defined policies, monitoring the job and updating job. Several software have been built on top on Dryad such as Dryad-LINQ and Microsoft server 2005 integration services.

2.3.1.3. Talend Open Studio. Talend Open Studio provides graphical interface to the users to visually analyze data. It is an open source software developed from Apache Hadoop. Unlike Hadoop, user can solve their problems without writing Java code. Moreover, users have this flexibility to drag and drop icons according to their defined tasks. Visual representation of components give users an ease to understand its working but at the same time it doesn't give much details to deeply understand the mechanism.

2.3.1.4. Apache Mahout. The Apache Mahout is a tool that addresses challenges that are associated with Big Data using machine learning techniques for data analysis. It is suited for large scale applications that require intelligent data analysis. Companies such as Facebook, Amazon, Twitter and Google have implemented and using machine learning algorithms to resolve their Big Data problems. Apache Mahout uses Hadoop and Map/Reduce platform. It uses number of well-designed and optimized algorithms such as clustering and classification, dimension reduction, pattern analysis etc.

2.3.1.5. Pentaho. Pentaho is one of the software platform that deals with business analytics, which handles structured and unstructured Big Data. Pentaho facilitates business professionals to serve with simplified access, better integration, visualization and better exploration of data. Moreover, it helps users to make decision based on data that will have an impact on the performance of the organization. Like other software such as JasperSoft, a chain of Pentaho's tools is developed and databases like Cassandra and MongoDB are connected with. Due to these connections with databases, users can easily access and retrieve information using web interface. It also provides web-based access to users that can easily analyze data and drill out decisions.

2.3.2. Stream processing tools

Stream processing means handling of large volume of real-time data. Applications such as sensors in industry, log files processing and online streaming require real-time processing of huge amount of data. The Big Data real-time processing requires very low latency while processing the data from large volume of data. The Map/Reduce model offers high latency as the Map phase data has to be save on disk before the reduce phase starts, which leads to high delay and makes it not feasible for real-time data processing. The big data in real-time stream processing faces some challenges when dealing with huge volume of data, high speed of data and the time dimension. To overcome the problems faced in Map/Reduce framework, real time platforms are introduced such as Storm, Splunk, SQLstream, S4, StreamCloud [18] and Apache Kafka. Few of them are discussed here.

2.3.2.1. Storm. The most popular platform for real-time stream processing is storm, which is open source, scalable, distributed and fault tolerant for unlimited streaming data. Storm is specifically developed for streaming data that is very easy to operate and ensures that whole data will be processed. Millions of records were processed per second on a single node that makes Storm an efficient platform for streaming data. In Storm platform, for different tasks different topologies are used while Hadoop creates Map/Reduce jobs for different applications. The key difference between Storm's topology and Map/Reduce job is that topology will process messages until user terminates it while job is finally dismissed. Topology is basically computational graph having two type of nodes namely spout and bolt. Starting point in the graph that represent the stream source is spout while bolt is responsible for processing input streams and then generating output streams. All nodes will have processing logic and links to represent the association between nodes. Storm platform has number of applications like online machine learning, real time analysis etc.

2.3.2.2. Splunk. Another intelligent and real time platform for accessing Big Data to obtain information generated by machines. It gives users the facility to access, monitor and analyze data through web interface. The results are represented through alerts, reports and graphs. Splunk Storm is the cloud version of Splunk. The characteristics of Splunk such as structured and unstructured data indexing, online searching, dashboard and real time reporting makes it different from other stream processing tools. Therefore, Splunk is a great application of log files. Moreover, it is used for diagnosing IT infrastructure problems and can be useful in business operations.

2.3.2.3. SQLstream. SQLstream is introduced to deal large scale real-time streaming Big Data. Its main focus is to discover patterns from unstructured huge amount of data such as patterns discovery in log files, data generated from different sensors and data from machines. The updated version is SQLstream s-Server that is developed to perform well in data gathering, conversion and sharing of real-time data. It is very much suitable for management and analysis of Big Data. Due to in-memory processing, it performs fast. The incoming data is considered as streams and is not stored on disk. By using SQL queries, it is processed in-memory as it arrives.

2.3.2.4. S4. S4 is another platform, which was initially released by Yahoo later on controlled by Apache. It is general purpose, fault-tolerant, distributed and expandable platform for unlimited streams of data in Big Data environment. In S4, programmers find it easy to develop applications as it is gives distributed environment, offers scalability and robustness and clusters can be easily managed. As S4 platform is implemented in Java with the aim that it should be modular that can easily process huge volume of streaming data. Like Storm, it also uses Apache ZooKeeper for cluster management. Due to its characteristics, S4 has been used in various applications such as Yahoo has used it for searching thousands of queries and has come up with good performance.

2.3.2.5. Apache Kafka. Initially, Apache Kafka was developed at LinkedIn for managing stream and operation data. It uses in-memory processing to obtain decision on real-time data. It is basically a distributed messaging system that offers, constant messaging, high throughput, distributed processing and loading parallel data. While extracting features from websites, the activity and operational data will play key role. Activity data is basically the actions of human on website such as copying contents, number of clicks and key words searching. Moreover, these activities are managed and aggregated for analysis or any decision making. While operation data is to measure the performance of machines (servers). Apache Kafka provides real-time processing for these two types of data.

2.3.3. Interactive analysis tools

In this era of Big Data, open source systems are developed to meet the requirements of not only batch and stream processing but also some platforms are introduced to deal with interactive processing. It allows user to interact with data and analyze information in their own way. In interactive processing, user can interact with computer in real time as they are directly connected to it. Moreover, it gives user the flexibility to view, compared and analyze data in graphical or tabular format or even both simultaneously. The Big Data tools that offer interactive analysis processing are briefly discussed here.

2.3.3.1. Google's Dremel. A well renowned company Google, proposed an architecture named as Dremel in 2010 that support interactive processing. Dremel's architecture is very different from Apache Hadoop that was developed for batch processing. Moreover, it has the ability to run bunch of queries in seconds over a table that contains trillion rows with the help of multi-level trees and column data. Furthermore, Dremel supports thousands of processors and has the capacity to accommodate petabytes of data of thousands of Google's users.

2.3.3.2. Apache Drill. A distributed platform to support interactive analysis processing of Big Data named as Apache Drill. It is more flexible than Google's dremel in terms of support for various query language, data types and different sources. Drill is the aim to support thousands of servers, to process petabytes of data and can process trillions of user records in few seconds. Dremel are Drill are designed to effectively explore the nested data. Both Google's dremel and Apache drill are specialists in large scale interactive analysis processing to respond to ad-hoc queries, as for storage they are using HDFS and for batch analysis, Map/Reduce model is used. It can scan data over petabytes in response to queries in few seconds, whether searching it in distributed file system or in column form. Google with its BigQuery offers Dremel-as-a-Service.

Please cite this article as: B. Jan et al., Deep learning in big data Analytics: A comparative study, Computers and Electrical Engineering (2017), https://doi.org/10.1016/j.compeleceng.2017.12.009

2.4. Big Data applications

The amount of data available in different field of life has increase the importance of Big Data to extract useful information for decision making. The traditional methods are not feasible for information extraction from huge volume of data. Therefore, deep learning plays a significant role in Big Data analytics and is widely used in different areas such as business operations, Smart city, Health care, Internet of Things, Social media, Recommendation system, Human behavior and many more. For instance, to handle the amount of data generated from social media and mobiles phones, deep learning techniques are used to extract information related to user activities.

3. Deep learning in Big Data analytics

The concept of deep learning is to dig large volume of data to automatically identify patterns and extract features from complex unsupervised data without involvement of human, which makes it an important tool for Big Data analysis [19]. In today's fast data growing world where huge amount of data having different formats and sizes are dealt with, different machine learning techniques along with computationally strong machines are required to deal with this volume and variety of data. Deep learning uses supervised/unsupervised techniques to automatically learn and extract data representations. It can be used to address Big Data problems (such as data tagging and indexing, information retrieval etc.) in a more efficient way, which is not possible with the traditional methods. Here we discuss two deep architectures of deep learning used with respect to its use in Big Data applications.

3.1. Deep Belief Networks and Big Data

The Deep Belief Network (DBN) has the capability to learn feature representation from labeled and un-labelled data. It constructs model by using unsupervised and supervised techniques. A DBN architecture consists of Input layer, hidden layers and output layer. Two directly connected layers forms restricted Boltzmann machine (RBM). Usually RBM has two layers, nodes in both layers are fully connected to each other and no connectivity between nodes within the same layer. The DBN's discrimination is used by many researchers to process big data in an efficient way. For instance, a Graphical Processing Unit (GPU) based architectural model is presented to process the massive amount of data in [20]. The author suggested the use of stacked Restricted Boltzmann Machines (RBM) in a parallel fashion to incorporate a huge amount of data with less processing time. Moreover, the proposed model process and train hundred million parameters compared to the previous research work where they process 3.8 million parameters [20]. However, implanting the proposed model for large-scale data have several problems such as transferring the data between client and global memory. One of the most efficient work has done in this regard by placing all the parameters and training solution in the global memory during the training phase. Moreover, a data parallel processing can be used to perform concurrent updates across each block of information.

However, the GPU implementation shows efficient result in the case of incorporating several million parameters in RBM. A number of 4.5 million parameters and 1 million examples is passed to RBM for testing. The speed of DBN learning is increased by a factor of 70 [20].

3.2. Convolutional Neural Networks and Big Data

A Convolutional Neural Network (CNN) has many hierarchical layers, consisting of feature maps layers and classification layers. Typically, CNN start with convolutional layer that accepts data from input layer as shown in Fig 3. The convolutional layer is responsible for convolution operations having few filter maps of same size. Moreover, output from this layer is forwarded to sampling layer which is responsible for reducing size of forthcoming layers. In CNN, a large number of deep learning methods are connected locally. Therefore, such type of networks is implemented on several hundred cores based on GPU implementation. The feature maps are assigned based on the blocks of information coming from the previous layer. However, it directly depends on the size of feature maps. A single block consists of several threads and each thread is attached to a single neuron. Similarly, the rest of the process is carried out by the convolution of neurons, activation, and summation. Finally, the output from above methods is stored in a global memory. This entire process follows a backward and propagation model for processing data efficiently. However, a single propagation doesn't generate good results, therefore, the parallelizing of propagation is carried out by pulling or pushing operations. Moreover, such process is affected by the border effects because the neurons in one layer may connect to a different number of neurons.

The data parallelism directly depends on the global memory and feature map size. The shared memory always affect the parallelism process. However, a circular buffer concept of the limited shared memory is suggested in [21]. The operation of convolution is performed by a thread in a parallel fashion and the results are written back to global memory. However, in the case of big data analytics limited shared memory still exists a challenging job. Therefore, researchers suggest a method of combining the convolution and sampling process in one step [21]. Thus, a back propagation is applied with storing the activities and error values in one step. A similar approach of using two GPUs, five convolution, and three classification layers is suggested for high-speed processing of large data in [22]. A layer wise processing is performed by assigning half of

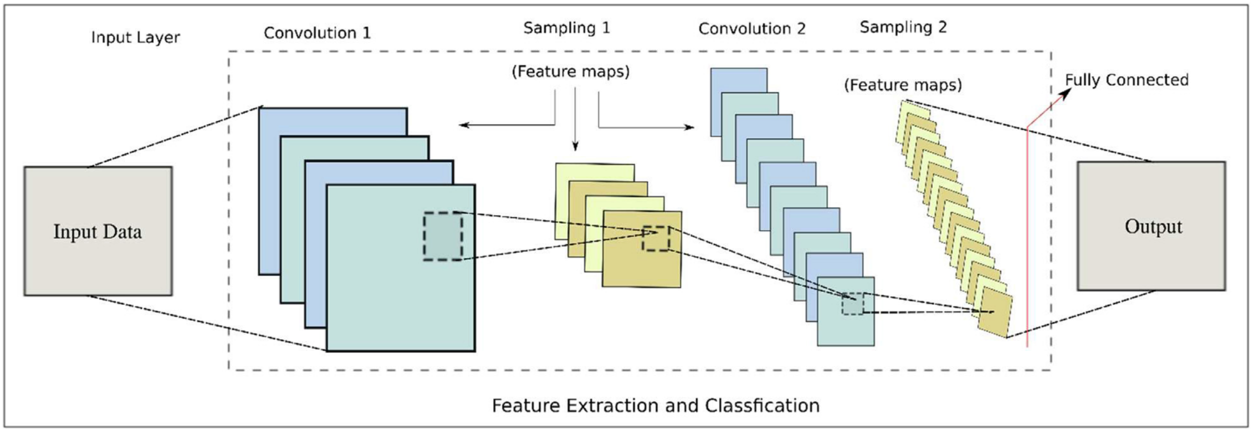


Fig. 3. CNN general architecture.

the layer processing to one GPU and half to another. Moreover, both the GPUs, transfer and communicate with each other without affecting host memory.

As mentioned earlier, a deep learning algorithm require preprocessing the input data, training the deep learning model, storing the trained model and then deployment of the model at last stage. In these steps, training the deep learning algorithms (define-and-run) is the most compute (or data) intensive task. In the define stage (also called forward pass) the model is given some input via a Neural network generating some output. In case of inappropriate or erroneous output the weights are re-adjusted (backward pass). This can be resembled to simple matrix multiplication where each input (row of first matrix) is multiplied with weight (column of the second matrix) for some specific output element. For higher order matrices (large inputs and weights) serial systems (CPU based) are usually not feasible. Fortunately, general purpose graphic processing units (GPGPU) provides far way better solution than traditional single or cluster CPU systems [23]. A number of high performance tools and frameworks have been developed to solve the computation intensive deep learning problems. Theano, CNTK, cuDNN, Chainer, COTS-HPC [24] etc provide better performance than CPU based system. However high speed up depends more on inherent parallelism of the algorithm than the number of GPU devices (multiple GPU systems) used in the computation. For instance, easy parallelization in single neural network could result in speedup gains by using multiple GPUs but not for dense neural network where parallelization is hard to achieve.

4. Deep Learning Techniques: A comparative approach

Neural Networks are used to process data in large volume in a faster and efficient way. Similarly, most of the research work proves that the efficiency and processing power of neural networks grows linearly and exponentially in its width and depth, respectively. For instance, if a Parity-N problem can be processed with SHN single hidden neural network with n neurons using following equation.

$$N = n - 1$$

Similarly, if the same problem is solved with a fully connected cascade architecture of FCC with same n neurons, the equation becomes

$$N = 2^n - 1$$

A more technical case is elaborated in Fig. 4 with a comparison of SHN and FCC for 15 neurons. The comparative study shows that the SHN can solve Parity-14 problem, however, the FCC can solve Parity-32,767 which is quite larger case.

Further, training a deep network with three hidden layers increases the difficulty level. Moreover, the difficulty level almost reaches to impossible solving when hidden layers reaches to 6 in number. One of the reason of such impossibility is the Vanishing Gradient Problem. Therefore, recently several approaches presented to tackle such situations using supervised and unsupervised methods. Recently most of the system are made automatic with the introduction of additional constraints in neural networks. The systems based on such phenomenon are called Bridged Multi-Layer Perceptron (BMLP) architectures. A comparative study of various neurons and their success rate in an MLP neural architecture is shown in Figs. 5 and 6 [24].

In [25], authors have used deep learning in Big Data for feature learning having different form of data. Tensor auto-encoder (TAE) is used to for features learning from heterogeneous data. To model the nonlinear relationship of data, authors have used tensor-based data representation. The tensor-auto-encoders are stacked to build tensor deep learning model to learn various levels of data representation. Moreover, tensor distance is used for learning set of features from huge amount of heterogeneous data. STL-10, CUAVE, SNAE2 and INEX 2007 representative classification datasets were considered to analyze the performance of the proposed tensor deep learning (TDL) model against Stacking Auto-Encoder (SAE). The classification accuracy was evaluated against the number of hidden layers using all datasets one by one. Table 1 shows the accuracy

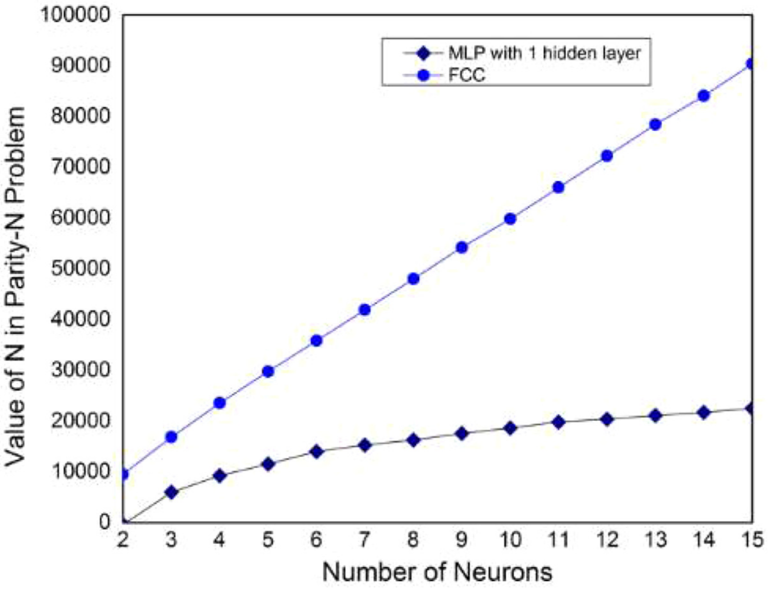


Fig. 4. Comparison of capabilities of SHN and FCC neural networks.

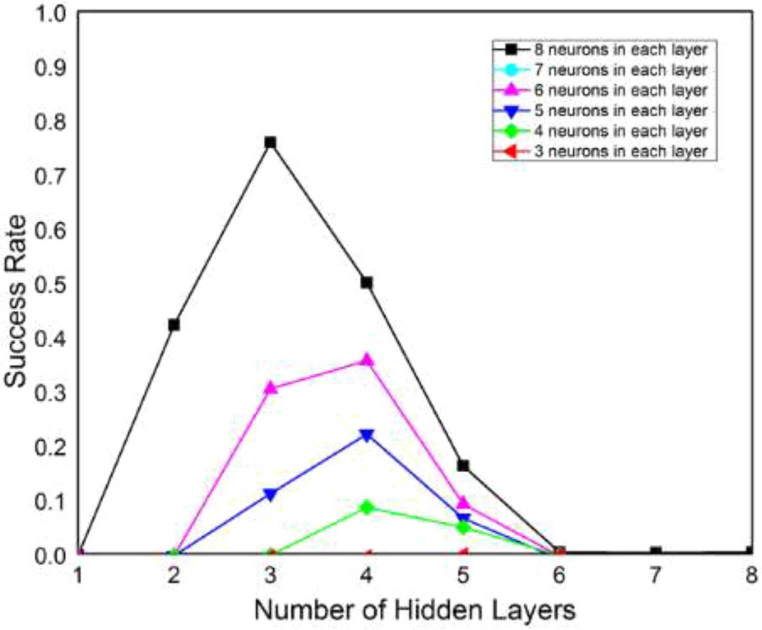


Fig. 5. Results of training 2 spiral problem with various MLP architecture.

Table 1
Summary of accuracy test.

	Test Accuracy (%)		
Datasets	SAE	TDL	MDL
STL-10	82 – 86	85 – 90	–
CUAVE	88.9 (Audio only) 67.7 (Video only)	91.6	89.1
SNAE2	–	85.7	81.4
INEX 2007	80.2	85.1	–

Please cite this article as: B. Jan et al., Deep learning in big data Analytics: A comparative study, Computers and Electrical Engineering (2017), https://doi.org/10.1016/j.compeleceng.2017.12.009

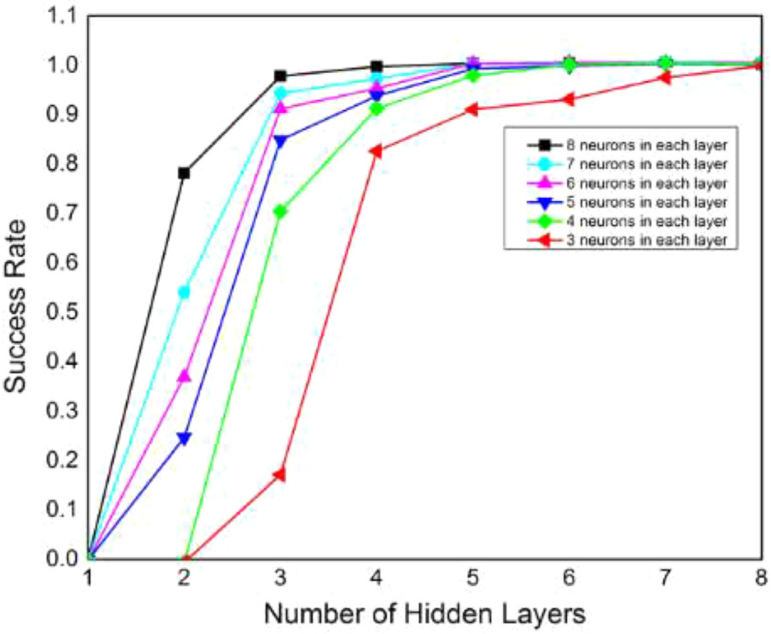


Fig. 6. Results of training 2 spiral problem with various BMLP architecture.

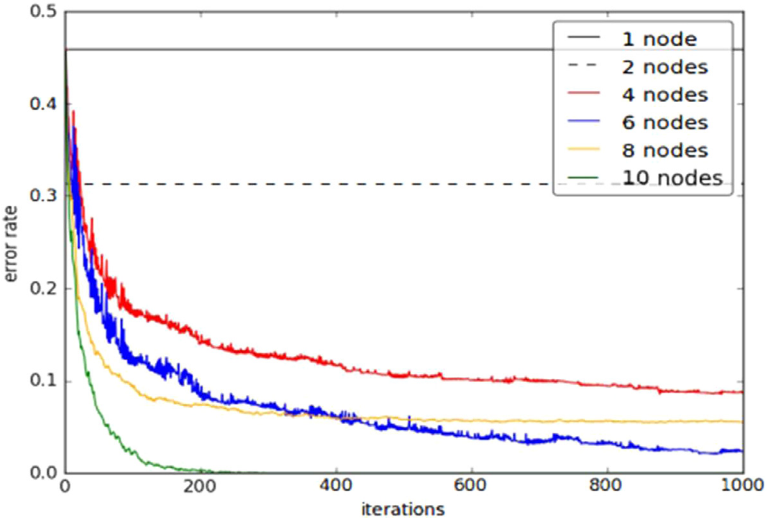


Fig. 7. Error rate vs number of iterations in terms of nodes, redrawn from [4]. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

summary of all four datasets against SAE, TDL and MDL. The results shows that the proposed TDL provides more accuracy as compared to other approaches.

The authors in [4] have used a distributed Deep Learning framework for sentiment analysis for Twitter users: as a typical Big Data application. Fig. 7 shows the error rate against the number of iterations for data of roughly over 160 million tweets towards sentiment analysis based on Deep learning for judgment of the writer and given textual tweet in accordance with a certain topic. The data, in the form of user tweets, was obtained using streaming API from Twitter Firehouse and partitioned in the ratio of 70/30 for training and test phases respectively. A distributed stochastic gradient method (SGD) was used in training phase. As shown in Fig. 7, the error rate of the model classifier decreases most often by increasing the number of nodes in a given number of iterations. Furthermore, for a certain number of nodes, the error rate of the classifier decreases by increasing the number of iterations. The process is performed for a number of iterations until the error rate drops to a certain threshold such that the new unseen data can be recorded with a certain acceptance level. In the given figure, the

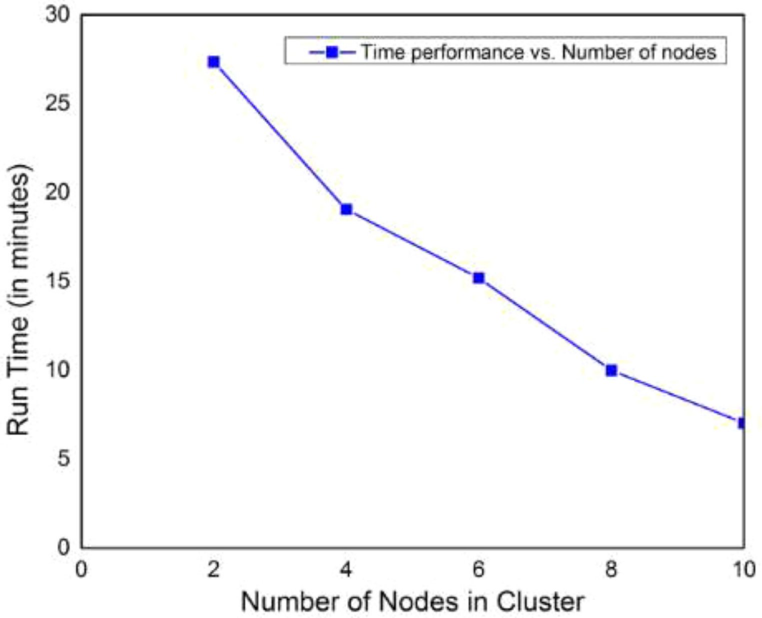


Fig. 8. Error rate vs number of iterations in terms of nodes.

bottom line (green line in color form) shows the convergence time of the model for 10 nodes in just 200 iterations. It can be stated from Fig. 8 that overall performance of the model is directly related to the number of nodes used.

5. Conclusion

In this article, we presented a comparative study of various deep learning techniques used for big data analytics. Moreover, a detailed classification of big data tools used for specific scenarios are shown. The deep learning techniques are broadly classified for big data learning and training, based on deep belief networks and convolution neural networks. The deep learning techniques have several limitations in processing big data with existing techniques. One of the major reason is processing the big and large amount of data in vector space. However, several sophisticated and optimized algorithms exist which can process data with high speed and classify feature learning with high accuracy. The comparative study is further elaborated by testing it on different data sets with a number of neurons and hidden layers. However, this study reveals, if the number of hidden layers increases from six, then it is impossible to solve a deep learning process.

Acknowledgments

This work (Grants No. C05407260100474320) was supported by Business for R&D funded The Korea Ministry of SMEs(Small and Medium Enterprises) and Startups in 2017, and by the National Research Foundation of Korea(NRF) Grant funded by the Korean Government (2015R1D1A1A01058171).

References

[1] Chen X-W, Lin X. Big data deep learning: challenges and perspectives. IEEE Access 2014;2:514–25.
[2] Zhou L, Pan S, Wang J, Vasilakos AV. Machine Learning on Big Data: Opportunities and Challenges. Neurocomputing 2017.
[3] Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E. Deep learning applications and challenges in big data analytics. J Big Data 2015;2:1.
[4] Khumoyun A, Cui Y, Hanku L. Spark based distributed Deep Learning framework for Big Data applications. In: Information Science and Communications Technologies (ICISCT), International Conference on; 2016. p. 1–5.
[5] Dumbill E. What is big data. O'Reilly media Inc. Big data Now: current perspectives. California: O'Reilly Media; 2012.
[6] Domingos P. A few useful things to know about machine learning. Commun ACM 2012;55:78–87.
[7] Zheng Y. Methodologies for cross-domain data fusion: An overview. IEEE Trans Big Data 2015;1:16–34.
[8] Qiu T, Zhang Y, Qiao D, Zhang X, Wymore ML, Sangaiah AK. A Robust Time Synchronization Scheme for Industrial Internet of Things. IEEE Trans Industr Inform 2017.
[9] Chen C, Liu X, Qiu T, Liu L, Sangaiah AK. Latency estimation based on traffic density for video streaming in the internet of vehicles. Comput Commun 2017;111:176–86.
[10] Agnihotri N, Sharma AK. Proposed algorithms for effective real time stream analysis in big data. In: Image Information Processing (ICIIP), 2015 Third International Conference on; 2015. p. 348–52.
[11] Simmonds RM, Watson P, Halliday J, Missier P. A Platform for Analysing Stream and Historic Data with Efficient and Scalable Design Patterns. In: Services (SERVICES), 2014 IEEE World Congress on; 2014. p. 174–81.

[12] Farman H, Javed H, Jan B, Ahmad J, Ali S, Khalil FN. Analytical network process based optimum cluster head selection in wireless sensor network. PloS One 2017;12:e0180848.
[13] Ding X, Tian Y, Yu Y. A real-time big data gathering algorithm based on indoor wireless sensor networks for risk analysis of industrial operations. IEEE Trans Industr Inform 2016;12:1232–42.
[14] Wu F, Wang Z, Zhang Z, Yang Y, Luo J, Zhu W. Weakly semi-supervised deep learning for multi-label image annotation. IEEE Trans Big Data 2015;1:109–22.
[15] Yang Y, Shen F, Shen HT, Li H, Li X. Robust discrete spectral hashing for large-scale image semantic indexing. IEEE Trans Big Data 2015;1:162–71.
[16] Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters. Commun ACM 2008;51:107–13.
[17] Pavlo A, Paulson E, Rasin A, Abadi DJ, DeWitt DJ, Madden S. A comparison of approaches to large-scale data analysis. In: Proceedings of the 2009 ACM SIGMOD International Conference on Management of data; 2009. p. 165–78.
[18] Gulisano V, Jimenez-Peris R, Patino-Martinez M, Soriente C, Valduriez P. Streamcloud: An elastic and scalable data streaming system. IEEE Trans Parallel Distribut Syst 2012;23:2351–65.
[19] Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives. IEEE Trans Pattern Analysis Mach Intell 2013;35:1798–828.
[20] Raina R, Madhavan A, Ng AY. Large-scale deep unsupervised learning using graphics processors. In: Proceedings of the 26th annual international conference on machine learning; 2009. p. 873–80.
[21] Scherer D, Müller A, Behnke S. Evaluation of pooling operations in convolutional architectures for object recognition. In: Artificial Neural Net-worksâ€“ICANN 2010; 2010. p. 92–101.
[22] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems; 2012. p. 1097–105.
[23] Jan B, Khan FG, Montrucchio B, Chronopoulos AT, Shamshirband S, Khan AN. Introducing ToPe-FFT: An OpenCL-based FFT library targeting GPUs. Concurrency and Computation: Practice and Experience; 2017.
[24] J. Bergstra, O. Breuleux, P. Lamblin, R. Pascanu, O. Delalleau, G. Desjardins, I. Goodfellow, A. Bergeron, Y. Bengio, and P. Kaelbling, "Theano: Deep learning on gpus with python," 2011.
[25] Zhang Q, Yang LT, Chen Z. Deep computation model for unsupervised feature learning on big data. IEEE Trans Services Comput 2016;9:161–71.

Bilal Jan received the M.S. and Ph.D degrees from the department of control and computer engineering (DAUIN) Politecnico di Torino, Italy, in 2010 and 2015 respectively. He is currently working as Assistant Professor and HoD of the Department of Computer Science, FATA University, FR Kohat, Pakistan.

Haleem Farman received the M.S. degree from International Islamic University Islamabad Pakistan in 2008. He is currently doing his Ph.D. degree in Computer Science from the Department of Computer Science, University of Peshawar Pakistan. He is currently working as a Lecturer in the Department of Computer Science, Islamia College Peshawar, Pakistan.

Murad Khan received the B.S. degree in computer science from university of Peshawar Pakistan in 2008. He completed his Ph.D. degree in computer science and engineering from School of Computer Science and Engineering in Kyungpook National University, Daegu, Korea. Dr. Khan published over 40 International conference and Journal papers along with two book chapters in Springer and CRC press.

Muhammad Imran is an assistant professor and member postgraduate advisory committee at Department of Computer Science and Information Technology, Sarhad University, Pakistan. He received his PhD (with Distinction) in Computer Science (specifically in the area of cloud computing and data preservation) from University of Vienna, Austria in 2014. His research interests include Cloud Computing, Data Preservation, Provenance and Service-Oriented Architectures.

Ihtesham Ul Islam received the B.*Sc*. degree from the University of Engineering and Technology of Peshawar, Pakistan, in 2006 and M.S. degree from Myongji University, South Korea, in 2009. He did a PhD from the University of Politecnico di Torino, Itlay in the year 2015. Currently he is an Assistant professor in Sarhad University of Science and IT, Peshawar, Pakistan.

Awais Ahmad received his Ph.D. in Computer Science and Engineering from Kyungpook National University, Daegu, Korea. He is currently working as an Assistant Professor in the Department of Information and Communication Engineering, Yeungnam University. Since 2013, he has published more than 70 International Journals/Conferences/Book Chapters in various reputed IEEE, Elsevier, and Springer Journals whereas in leading conferences.

Shaukat Ali is a PhD scholar in the Department of Computer Science, University of Peshawar-Pakistan. He received his MSc and MS degrees from the Same University of Peshawar in 2007 and 2010 respectively. He is working as a lecturer in the Department of Computer Science, Islamia College Peshawar-Pakistan. His area of interest is data security and user privacy.

Gwanggil Jeon received the BS, MS, and PhD degrees from Hanyang University in 2003, 2005, and 2008. Then, he was with the University of Ottawa, as a postdoctoral fellow. From 2011 to 2012, he was with the Niigata University, as an assistant professor. He is currently an associate professor with the Department of Embedded Systems Engineering, Incheon National University.

Please cite this article as: B. Jan et al., Deep learning in big data Analytics: A comparative study, Computers and Electrical Engineering (2017), https://doi.org/10.1016/j.compeleceng.2017.12.009