JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

IMPLICATION OF DEEP LEARNING AND BIG DATA IN MEDICAL IMAGING

[1]A. Posi sekhar, [2]G. Naga Lakshmi, [3]L.Dasaradha Ramayya

[1]Asst.Prof, Computer Science Dept, GIET Degree college, Rajahmundry, AP, India.

[2]Asst.Prof, Computer Science Dept, GIET Degree college, Rajahmundry, AP, India.

[3]Associate.Prof, Computer Science Dept, ISTS WOMEN's Engineering college, Rajanagaram, AP, India.

Abstract

Big Data Analytics and Deep Learning are two high-focus of data science. Big Data has become important as many organizations both public and private have been collecting massive amounts of domain-specific information, which can contain useful information about problems such as national intelligence, cyber security, fraud detection, marketing, and medical informatics. In recent years, by collecting medical data of patients, converting them into Big Data and applying appropriate algorithms, reliable information has been generated that helps patients, physicians and stakeholders in the health sector to identify values and opportunities. With the new technologies such as Hadoop, it is now feasible to store and use extremely large volumes of data that comes in at an unprecedented velocity. This paper presents a brief overview of big data analytics in medical imaging with considering the significance of deep learning technique.

Keywords: Big Data, Deep Learning, Medical Imaging, Hadoop

Introduction

The continuous advancement in medicine, genome, pharmaceutical, and health care monitoring is a result of the development and application of technological devices. This has made it possible to easily capture data for analysis and processing. Similarly, improvement in technology also makes it possible to store very large amount of data with useful information.

Big data provides the opportunity for health policy experts, physicians, and health care institutions to make data-driven judgments that will enhance patient treatment, disease management, and health care decisions. Many experts have used internet tools for big data services and related applications. This is depicted in the graph in Figure 1, which was obtained from Google Trends for "big data in healthcare" between 2010 and 2018.

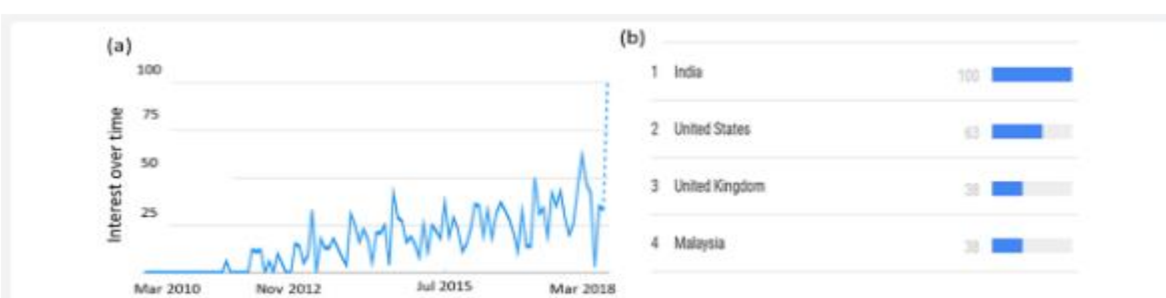

Figure 1 Google Trends for "big data in healthcare" between 2010 and 2018; (a) occurrence timeline graph and (b) prevalence occurrence by country

Google Trends is a free Web service by Google Inc that provides statistical occurrence of activities by people on the internet all over the world. The trend in the graph is calculated as interest over time on a scale from 0 to 100, where 100 refers to the maximum computed score for total search and related activity for the topic.

The size of medical data is too large for comprehensive analysis with the available analytical tools to maximize the knowledge available in big data. Traditional machine learning (ML) techniques and algorithms have limited capacity to utilize big data and, in most cases, the solution becomes complex and undesirable. Deep learning (DL) is proposed and provides a prospective solution to this challenge.

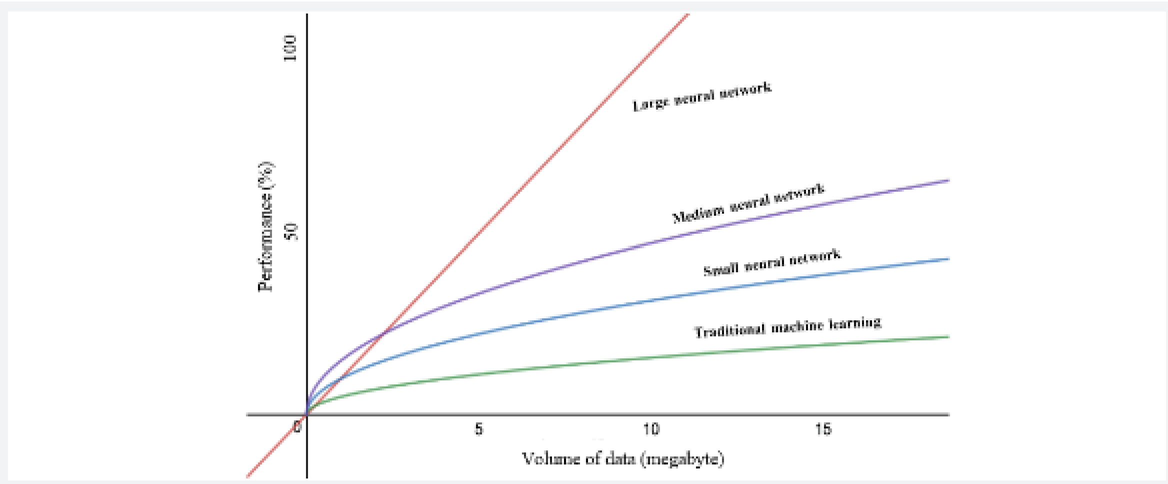

Figure 2 shows the performance between DL and other ML techniques in the situation of increasing data size. The primary advantage of DL is that the performance of large architecture of DL increases with increase size of available data

To help the accomplish task performed by humans and overcome these limitations, intelligence demonstrated by humans is built into machines and computers to create the concept of artificial intelligence (AI). ML is a branch of AI that gives computers the ability to learn and perform the role of experts without being explicitly programmed. Some examples of ML include support vector machine (SVM), decision tree, logistic regression, Naïve Bayes, K-means clustering, and so on. On the basis of a broad classification scheme, ML can be categorized into 3 groups.

The recent hunger for data consumption and analysis has opened up new frontier for more ideas and applications. Artificial neural network or neural network (NN) is another example of ML, with an interconnection of nodes called neurons with 3 major layers, input, hidden, and output layers, where the hidden layer is a single layer that connects the input layer to the output layer. The purpose of NN is to gradually approximate a function that maps an input to a corresponding output through an iterative optimization process. NN has transformed from its inception as a simple perceptron to solve simple problem to the advanced concept of deep neural network (DNN), which has many cascaded interconnected hidden layers that are able to process and analyze audios, text, signals, images, and more complex data types. DL is the recurrent learning process performed in DNN that enables it to find an optimal function for representing data. The innovation of DL is a developing trend in data analysis and is ranked as one of the best inventions in technologies. DNN is an active branch of ML and its goal is to make machines think and understand as humans by mimicking the grid of the human brain connection, to focus on learning data representation (DR) rather than task-specific algorithms.

Currently, DL has started making huge impact across different areas in health care. The increasing availability of health care data and rapid development of variations in DL techniques have made it possible to have the impressive results recorded in health care. DL techniques can reveal clinically relevant information hidden in large amount of health care data, which in turn can be used for decision making, treatment, control, and prevention of health conditions. Some application areas of DL include health behavior reaction, EHR processing and retrieving scientifically sound treatment from text, eye related analysis and classification, gait analysis and robotic-assisted recovery, hearing disorder treatment, cancer treatment, heart diagnosis, and brain activity analysis .

Big Data Fundamentals and Elements

During the past few years, BD technologies have been highlighted as a fundamental and strategic support for very different aspects such as productivity growth, innovation, or customer relation, to name a few. These in turn can strongly benefit large-scale business and sectors, like retailing, manufacturing, healthcare, the public sector, or entire cities .Probably the power of BD is not only in the current availability of extensive data volumes in the organizations, but also in the application of adequate and well-chosen analysis techniques. Working together, these two elements are expected to allow us to gain valuable insight in new applications and markets. According to Gartner's definition, BD analytics can actually provide us with three main benefits, namely, valuable insights can be extracted from data, better decisions can be made based on those insights, and decision-making can be automated and included in the management process itself.

Current definitions of BD are dependent on the techniques and technologies used to collect, store, process, and analyze available data . BD solutions have been established to usually share several principles. First, they often use high-level architectures, which are necessary to address the different and specific underlying data technologies and sources. Second, they usually include a variety of Data Science tasks, such as Data Mining, Statistical Analysis, ML, Real-Time Visualization, or In-Memory Analysis. Third, they combine the benefits of different tools for different tasks. Fourth, distributed processing and storage is often used across different nodes in a cluster. In addition, finally, coordination among data and processing nodes is ensured to improve scalability, efficiency, and fault-tolerance.

figure 3 applications of big data in health care

The following paragraphs outline the most relevant techniques and technologies which are used today to carry out the design and implementation of BD solutions, as they can roughly be grouped into considerations about the BD life cycle, the infrastructural requirements, the frameworks and libraries, and the visualization tools.

(1) The BD Life Cycle. Dealing with BD implies gathering related knowledge in areas like ML, programming, and databases. Techniques such as sentiment analysis, time series analysis and forecasting, spatial analysis, optimization, visualization, or unstructured analytics (namely, audio and video) are necessities in the BD life cycle for extracting value from it , especially when performed in a distributed manner to achieve scalability. This BD life cycle is significantly different from traditional environments, in which data are first explored, then a model is designed, and finally a database structure is created. In BD environments, however, data are first collected and loaded to a certain storage system, then a metadata layer is applied, and finally a structure is created. Therefore, data are first transformed and then loaded in traditional approaches, whilst data are first loaded and then transformed in BD approaches .

(2) Infrastructure Requirements. An adequate infrastructure is fundamental for providing BD solutions. Among the most relevant are the cloud models, such as the increasingly widespread Infrastructure as a Service (IaaS). In these kinds of models, commodity hardware plays a relevant role, especially due to the lower costs in building shared-nothing architectures. Two BD architectures stand out among so many, namely, the Lambda Architecture and the National Institute of Standards and Technology (NIST) BD Reference Architecture . In order to satisfy particular needs, Lambda Architecture uses a system based on three main components, namely, a batch layer, a serving layer, and a speed layer (for real-time data). However, NIST BD Reference Architecture is rather an open tool (not tied to specific vendors, services, implementations, or any specific solutions) to facilitate the discussion of requirements, structure designs, and operations that are inherent in BD environments.

(3) BD Frameworks and Libraries. A growing number of frameworks and software libraries can be found to support BD development. MapReduce is an algorithm for processing parallelizable tasks across large datasets using a great number of nodes, which has been fundamental for the advent of the BD scenarios. The most commonly recognized open-source implementation of MapReduce is Apache Hadoop, which is capable of providing reliable and scalable distributed computing. In addition to being open source, Hadoop is less expensive, it has a more flexible storage (mainly in unstructured data), it is adequate for massive scans, and it has deep support for complex structures. Beyond the processing capabilities of MapReduce, another option for large scale number crunching is General Purpose GPU Programming (GPGPU), such as Nvidia CUDA and OpenCL framework. Another solution, based in Hadoop, is Mahout

library , which is an ML and Statistics library able to work in distributed application environments. However, the MapReduce key-value approach prevents the implementation of many ML algorithms, and it is too often only useful for an initial phase of data preprocessing rather than for an advanced ML-based data analysis. For this reason, Mahout is not very scalable in terms of most ML algorithms. Apache Spark is a unified analytics engine for large-scale data processing, which is certainly more flexible than MapReduce and Mahout, but it still has some limitations when typical ML operations, such as $\mathcal{O}(n3)(3)$-matrix inverse calculations or decompositions, are implemented. Sci DB is a column-oriented database management system designed for multidimensional data management and analytics, which is an attempt to unify the relational databases in a multidimensional way, trying to overcome some of the limitations that the relational databases have with BD problems. Graph database libraries like Neo4j or Titan are also widely used in BD, since they make use of graph theory for high-density operations of traversing and querying data without expensive requirements nor index lookups . Moreover, graph databases store and access data by using the same abstraction are used to explain and to structure them. Pregel , Apache Giraph, and Apache Hama are implementations of the Bulk Synchronous Parallel (BSP) model, which is a bridging model for creating parallel algorithms . BSP works in a similar way to a communication network where each individual processor has its private, fast, and local memory. This makes BSP highly effective at processing large amounts of highly interrelated information. Other interesting libraries related to BD are: MLPACK , a state-of-the-art, scalable, multiplatform C++ ML library that provides cutting e.g., algorithms whose benchmarks exhibit far better performance than other leading ML libraries; in addition, GraphLab, a Python ML library designed to operate on either a single machine or an existing distributed cluster.

The above solutions are only a fraction of what is available under the heading of BD. Each one has been created with a specific problem subset in mind, and hence with its advantages and disadvantages. After analyzing these solutions with respect to the four V's of BD (Volume, Velocity, Variety, and Variability), it was concluded that the best solution depends: (1) On how our data are structured; (2) On the specific problem we are trying to solve; (3) On the hardware we have available; (4) On the experience, knowledge, and skills that are available; (5) On the amount of time we can have; (6) On the algorithm or algorithms that we need to utilize; (7) If we need to respond to immediate requests for results or scheduled batches; (8) If we have static or shifting data; (9) Or if we are utilizing existing storage or need a new system. Authors of BD State-of-the-Art in point out that "it is vital that you examine your own goals and ruthlessly parse the available options with an eye for what you are trying to achieve, but also what you might wish to achieve in the future".

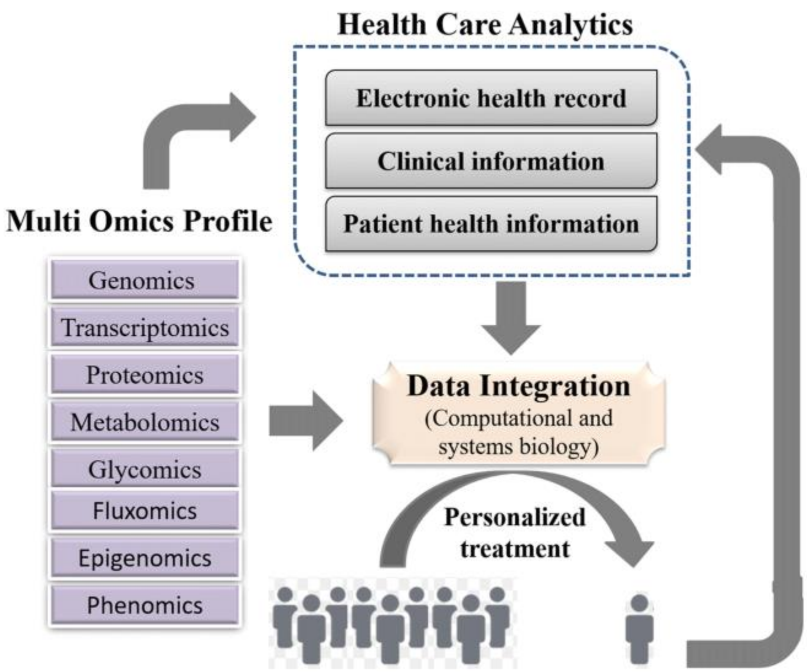

figure 4 leveraging big data analytics in health care

(4) Visualization. After analyzing data, the results need to be visualized through a good graphical tool. Regarding BD visualization, several tools can be highlighted, such as Datameer, FICO BD Analyzer (former Karmasphere), Tableau, or TIBCO Spotfire . Other solutions used to visualize the results include statistical tools (such as R or SAS), business intelligence tools (like Jaspersoft Business Intelligence Suite or Pentaho Business Analytics), programming languages (such as JavaScript or Python), and office tools (like Excel) .

Deep Learning Fundamentals and Elements

Artificial Intelligence (AI) could be defined as the capacity of computers to exhibit or simulate intelligent behavior when automating tasks that are normally performed by humans . AI was born in the 1950s, when researchers in the field of computer science started asking whether computers could be made to think .In the beginning, researchers believed that AI could be achieved with handcrafted programs with a large set of explicit predefined coding rules specified by human operators. This kind of AI is known as symbolic, and it was the dominant paradigm in AI from the 1950s until the late 1980s. Even though this paradigm is useful at solving well-defined logical problems like playing chess, it is incapable of solving complex problems where finding out explicit rules is very hard, for instance, image classification, object detection, speech recognition, or language translation. However, ML emerged as a new approach replacing symbolic AI and it started to grow quickly in the 1990s, until becoming probably the most popular and most successful field of AI.

Broadly speaking, ML can be seen as the ability of computers to extract information from the input of raw data and to learn from experience, thus generating complex inferences based on the relationships within the data . Thereby, ML builds predictive models without using predefined coding rules , and it is able to deal with large and complex datasets for which statistical analysis would be unfeasible. We can divide ML algorithms into three categories, namely, supervised, unsupervised, and reinforcement learning. In supervised learning, the goal is to learn a function that, from a sample of input–output pairs (labeled as training data), approximates the relationship among inputs and outputs well enough in the labeled training data, and then the inferred function can be used for mapping new data. By contrast, unsupervised learning seeks to identify hidden patterns present in datasets, without the requirement that they are labeled. Lastly, reinforcement learning aims to maximize the accuracy of algorithms learning actions that maximize some reward .

ML algorithms strongly depend on feature engineering, which is the process of using domain knowledge of the data to create features that make these algorithms work. The process requires lots of time and effort for feature selection, and features must extract relevant information from huge and diverse data in order to produce the best outcome. These approaches have shown good results on structured data, such as sales predictions or recommendation systems; however, in unstructured domains, such as computer vision and natural language processing, feature engineering is a challenging task for ML algorithms . To handle these problems, a special branch of ML arose, so-called DL, which is based on the concept of Artificial Neural Networks (ANNs, also known simply as Neural Networks) . The major difference between DL and traditional ANNs are the number of hidden layers. ANNs are usually limited to three layers and they are known today as Shallow Neural Networks, whereas DL are NNs with many layers and they are referred to as Deep Neural Networks (DNNs). DNNs use multiple layers to explore more complex nonlinear, patterns and learn meaningful relationships within the data , and they learn and construct inherent features from each successive hidden layer of neurons, by minimizing or even removing the need for feature engineering. This last factor resulted in DL often outperforming ML techniques, revolutionizing this field with outstanding results and robustness to input noise and variability in diverse tasks .

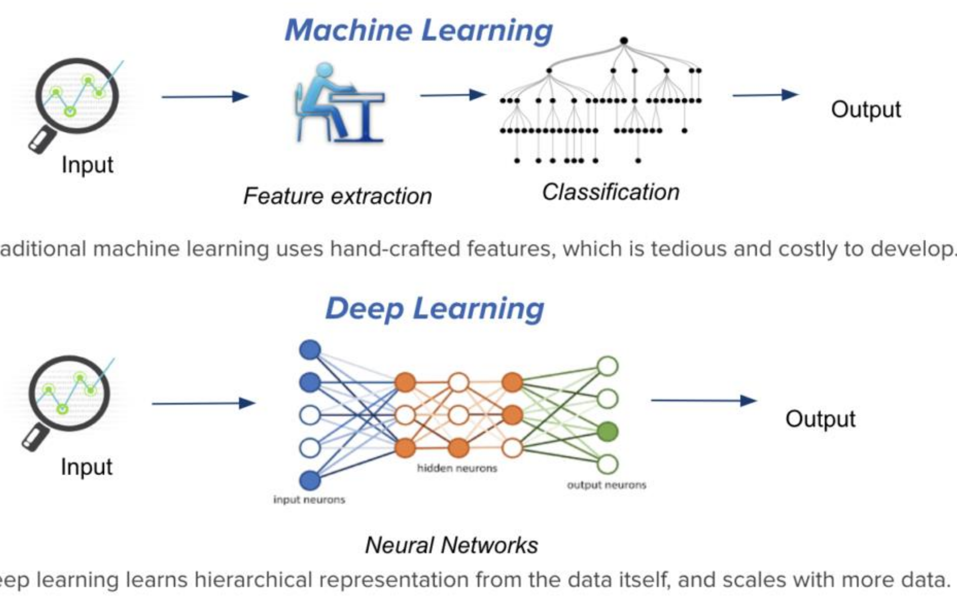

figure 5 machine learning vs deep learning

Therefore, some authors point that DL could be seen as a specific case of BD solution, since the implementation of BD techniques, such as the parallelization in GPUs, has made this computationally demanding solution viable. This view is not shared by everyone, as the BD regime is often assumed to be fast changing and moving with time. Nevertheless, the previously discussed different and changing views of BD definitions are also shared by other authors. In any case, DL needs to overcome several challenges before becoming a widespread solution, including the requirement of large training sets and the demand for a tremendous computational power, both of them in order to be able to train the many hidden layers. The previous causes can produce a lack of generalization in the solution with new data or overfitting. The lack of ability to explain or interpret the reasons for the obtained output (black box) is also an open issue. The advent of BD with very large datasets and GPUs with hardware improvements have enabled to reduce the computational problem. In addition, research efforts in recent years are opening new ways of addressing the problems of overfitting and interpretability. There have been other major issues such as better easy-to-use software libraries (like Theano, TensorFlow, Keras, or Pytorch), better algorithms, and a large community of researchers and developers, which when put together have had an incredible impact on our world and reached a level of public attention and industry investment never seen before in the history of AI .

Since DL is based on the concept of ANNs, we start with a brief history of the evolution of its algorithms. Artificial neurons were first proposed in 1943 as a model for how neurons process information in the biological brain. Early ANN algorithms tried to model the functionality of a biological brain with a structure that was made of three layers (input, hidden, and output). Each layer consisted of several artificial neurons based on activation functions which linked them to each other and linked to the next layer through weighted connections. Thereby, the axon is played by the output, the dendrites are played by the inputs, the nucleus is played by the activation function, and the synapses are played by the weights. ANNs are especially useful in nonlinear, classification problems. After the 1960s, research in ANNs slowed down due to the low capability, their shallow structure, and the limited computational capacity of computers. Some papers in the 1980s significantly contributed to the comeback of ANN algorithms with the emergence of back propagation , an algorithm based on the chain rule that made training possible, so that weights were modified by evaluating the difference between the predicted and the true class labels. However, back propagation seemed to work only for certain types of shadowed ANNs , and training a mostly deep ANN (with more layers) was difficult in practice , mainly due to high computational complexity and the limited size of the databases. Other ML algorithms, such as Support Vector Machines (SVMs), Random Forest, and k-Nearest Neighbors (kNN) algorithms, gradually surpassed ANNs in popularity . In 2006, there was a real impact milestone, when a paper entitled “A fast learning algorithm for deep belief nets” was published. The idea that this work proposed was that DNNs with many layers can be trained if the weights are initialized in the correct way. This point is considered the beginning of the DL movement. Since then, more important DL papers started to appear. The 2007 paper entitled “Greedy Layer-Wise Training of Deep Networks” concluded that DNNs are more efficient for difficult problems than other ML algorithms (included shadow NN). However, by 2010, DL was being almost completely ignored by the scientific community, except for some groups: Geoffrey Hinton at the University of Toronto, Yoshua Bengio at the University of Montreal, Yann LeCun at New York University, and IDSIA in Switzerland . An important breakthrough came about in 2012, when Hinton and his group were able to increase the top-5 accuracy from 74.3% to 83.6% at the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Since then, DL has become the reference algorithm for a large number of tasks, like computer vision and natural language processing, both in the scientific and in the industrial world .

DL, just like ML, can be used both in supervised applications as well as in unsupervised applications. Depending on this application, DNN architecture comes in many different forms , which can be grouped into three general families.

(1) Feed-Forward Neural Networks (also known as Multi-Layer Perceptron, MLP). This architecture can be considered as the par excellence DL model. It consists of artificial neurons grouped in an input layer, many hidden layers, and an output layer, in which each previous layer is weight connected via some activation function to the next layer. Thereby, information flows from the input layer to output layer without feedback connections .

(2) Convolutional Neural Networks (CNN). Feed-Forward Neural Network architecture performance is affected by translation and shift deviation, which are pernicious for some tasks, like those that are related to images. CNN were created to eliminate these deficiencies while accounting for the properties of translation and shift invariance. This DL architecture is one of the most popular and the most influential innovations in the field of computer vision. Inspired by the neurobiological architecture of the visual cortex (cells in the visual cortex are sensitive to small regions of the visual field), it was first proposed and advocated for image analysis by Lecun et al. in 1989. It is a hierarchical model that consists of convolutional and subsampling layers. In the convolutional layer, artificial neurons compute a spatial convolution extracting features of small portions of input images during the training. A special CNN model (AlexNet) was used by Hinton’s group to win ILSVRC in 2012.

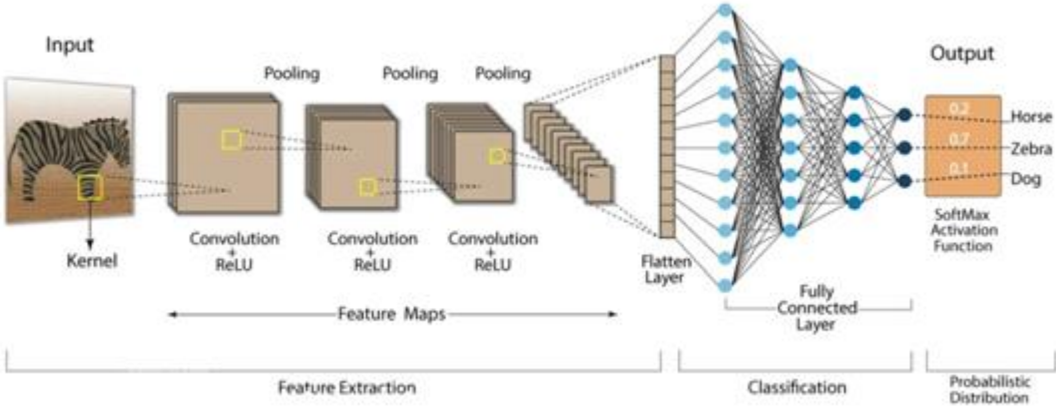

figure 6 convolution neural network architecture

(3) Recurrent Neural Networks (RNN). This architecture is a variation of Feed-Forward Neural Networks that exhibits dynamic behavior. Artificial neurons are connected to each other with a delay and are associated with time steps, in a way that parameters across this different time steps are shared. Basically, the RNN architecture receives the input, it updates its high-dimensional hidden state information that is dependent on the previous computation, and it makes a prediction at each time step. This approach has become the primary tool for handling sequential data because it outperforms other DL approaches in dealing with time-variant data. It has been applied successfully in natural language processing, speech recognition, and machine translation. To overcome the vanishing gradient, Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) RNN architectures are frequently used, since they incorporate memory blocks with gates that control the state and decide which information to add or to remove in each step.

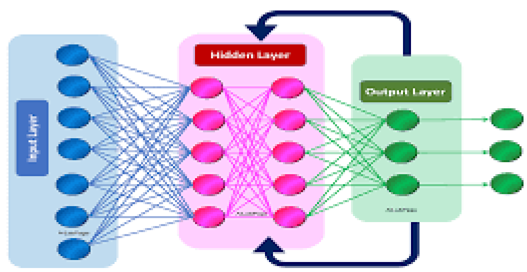

figure 7 recurrent neural networks architecture

BD has not yet started to be widely used in some disciplines, and sometimes there are very limited databases in them. This lack of data to train DL architectures can produce a problem called overfitting, which occurs when a learning model memorizes the training data and it does not generalize well when faced with new data. Even when large datasets are available, the complexity of DL models can make DNNs more prone to overfitting. A common way to avoid overfitting is to simplify the DNN architecture by forcing its weights to take only small values. This method is called weight regularization, and it can be said to have two main flavours, known as $L1$ and $L2$ regularization. Another plan is to use Dropout, which is one of the most effective and most commonly used regularization techniques for NN. It consists of randomly setting to zero a number of artificial neurons of the layer during training . Apart from regularization, data augmentation is another important way to avoid overfitting. It refers to increasing the number of data points applying transformation to the dataset. It is normally used with images, by rotating different angles or adding different levels of noise, cropping, and geometrically transforming (e.g., rotation, translation, and scaling).

Conclusion

This paper provides brief introduction to big data and its applications. The major intention of this study is to propose an optimized and secure big data healthcare framework. It is observed from the existing literature that the Machine Learning based Application Layer Health Information Services, Epidemic Outbreak ,forecasting Disease Diagnosis, Drug Discovery, New Healthcare applications of big data in healthcare industry revolutionize medical industry by providing better health and information to patients. Moreover, the use of information technology assist in reducing the costs associated with healthcare diagnosis. Here, initially the big data concept along with its major applications has been briefly introduced. In this review, we have completed an exploration of the current state of BD and DL with the objective of providing a snapshot of these two branches of Data Science. In the first parts of the work, fundamentals of both BD and DL were presented together with the most relevant techniques and technologies used today in these fields. Then, some of the success stories in technological areas like computer vision and natural language processing, in finance, and other relevant areas such as agriculture, transportation, marketing and advertising, and education have been overviewed. A special interest has been placed on applications in healthcare

References

1. Kelleher, J.D.; Tierney, B. Data Science, 1st ed.; MIT Press: Cambridge, MA, USA, 2018. [**Google Scholar**]
2. Sorescu, A. Data-Driven Business Model Innovation. J. Prod. Innov. Manag. **2017**, 34, 691–696. [**Google Scholar**] [**CrossRef**]
3. Hoffmann, A.L. Making Data Valuable: Political, Economic, and Conceptual Bases of Big Data. Philos. Technol. **2018**, 31, 209–212. [**Google Scholar**] [**CrossRef**]
4. Press, G. 12 Big Data Definitions: What's Yours? Available online: **https://www.forbes.com/sites/gilpress/2014/09/03/12-big-data-definitions-whats-yours/** (accessed on 19 January 2019).
5. Goodfellow, I.; Bengio, Y.; Courville, A. Deep Learning, 1st ed.; The MIT Press: Cambridge, MA, USA, 2016. [**Google Scholar**]
6. Provost, F.; Fawcett, T. Data Science and its Relationship to Big Data and Data-Driven Decision Making. Big Data **2013**, 1, 51–59. [**Google Scholar**] [**CrossRef**]
7. Granville, V. Difference Between Machine Learning, Data Science, AI, Deep Learning, and Statistics. Available online: **https://www.datasciencecentral.com/profiles/blogs/difference-between-machine-learning-data-science-ai-deep-learning** (accessed on 19 January 2019).
8. Franks, B. Understanding Operational Analytics. In The Analytics Revolution; John Wiley & Sons, Ltd.: Indianapolis, IN, USA, 2014; pp. 1–31. [**Google Scholar**]
9. Madden, S. From Databases to Big Data. IEEE Int. Comput. **2012**, 16, 4–6. [**Google Scholar**] [**CrossRef**]
10. Dutcher, J. What is Big Data? Available online: **https://datascience.berkeley.edu/what-is-big-data/** (accessed on 19 January 2019).
11. Beyer, M.A.; Laney, D. The Importance of 'Big Data': A Definition. Available online: **https://https://www.gartner.com/doc/2057415/importance-big-data-definition** (accessed on 19 January 2019).
12. Laney, D. 3D Data Management: Controlling Data Volume, Velocity, and Variety. Available online: **http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf** (accessed on 19 January 2019).
13. Loshin, D. Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph, 1st ed.; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2013. [**Google Scholar**]
14. Sivarajah, U.; Kamal, M.M.; Irani, Z.; Weerakkody, V. Critical Analysis of Big Data Challenges and Analytical Methods. J. Bus. Res. **2017**, 70, 263–286. [**Google Scholar**] [**CrossRef**]
15. Costa, C.; Santos, M.Y. Big Data: State of the Art Concepts, Techniques, Technologies, Modeling Approaches and Research Challenges. IAENG Int. J. Comput. Sci. **2017**, 44, 285–301. [**Google Scholar**]
16. Chandarana, P.; Vijayalakshmi, M. Big Data Analytics Frameworks. In Proceedings of the 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), Mumbai, India, 4–5 April 2014; pp. 430–434. [**Google Scholar**]
17. Gandomi, A.; Haider, M. Beyond the Hype: Big Data Concepts, Methods, and Analytics. Int. J. Inf. Manag. **2015**, 35, 137–144. [**Google Scholar**] [**CrossRef**]
18. Krishnan, K. Data Warehousing in the Age of Big Data, 1st ed.; Morgan Kaufmann: Boston, MA, USA, 2013. [**Google Scholar**]
19. Cao, C.; Liu, F.; Tan, H.; Song, D.; Shu, W.; Li, W.; Zhou, Y.; Bo, X.; Xie, Z. Deep Learning and Its Applications in Biomedicine. Genom. Proteom. Bioinform. **2018**, 16, 17–32. [**Google Scholar**] [**CrossRef**]
20. McBee, M.P.; Awan, O.A.; Colucci, A.T.; Ghobadi, C.W.; Kadom, N.; Kansagra, A.P.; Tridandapani, S.; Auffermann, W.F. Deep Learning in Radiology. Acad. Radiol. **2018**, 25, 1472–1480. [**Google Scholar**] [**CrossRef**][**Green Version**]
21. Ching, T.; Himmelstein, D.S.; Beaulieu-Jones, B.K.; Kalinin, A.A.; Do, B.T.; Way, G.P.; Ferrero, E.; Agapow, P.M.; Zietz, M.; Hoffman, M.M.; et al. Opportunities and Obstacles for Deep Learning in Biology and Medicine. J. R. Soc. Interface **2018**, 15, 20170387. [**Google Scholar**] [**CrossRef**]
22. Ganapathy, N.; Swaminathan, R.; Deserno, T.M. Deep Learning on 1-D Biosignals: A Taxonomy-based Survey. Yearb. Med Inf. **2018**, 27, 98–109. [**Google Scholar**] [**CrossRef**]
23. Chen, M.; Mao, S.; Liu, Y. Big Data: A Survey. Mob. Netw. Appl. **2014**, 19, 171–209. [**Google Scholar**] [**CrossRef**]
24. Xiaomeng, S. Introduction to Big Data. Available online: **https://www.ntnu.no/iie/fag/big/lessons/lesson2.pdf** (accessed on 19 January 2019).
25. Marz, N.; Warren, J. Big Data: Principles and Best Practices of Scalable Realtime Data Systems, 1st ed.; Manning Publications Co.: Greenwich, CT, USA, 2015. [**Google Scholar**]
26. Chen, C.L.P.; Zhang, C.Y. Data-Intensive Applications, Challenges, Techniques and Technologies: A Survey on Big Data. Inf. Sci. **2014**, 275, 314–347. [**Google Scholar**] [**CrossRef**]
27. Garber, L. Using In-Memory Analytics to Quickly Crunch Big Data. Computer **2012**, 45, 16–18. [**Google Scholar**] [**CrossRef**]
28. Kimball, R.; Ross, M. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd ed.; John Wiley & Sons, Ltd.: Indianapolis, IN, USA, 2013. [**Google Scholar**]
29. NIST Big Data Public Working Group. NIST Big Data Interoperability Framework: Volume 6, Reference Architecture. Available online: **https://bigdatawg.nist.gov/_uploadfiles/NIST.SP.1500-6r1.pdf** (accessed on 19 January 2019).
30. Apache Software Foundation. Welcome to Apache Hadoop! Available online: **https://hadoop.apache.org** (accessed on 19 January 2019).

31. Bakshi, K. Considerations for Big Data: Architecture and Approach. In Proceedings of the 2012 IEEE Aerospace Conference, Big Sky, MT, USA, 3–10 March 2012; pp. 1–7. [**Google Scholar**]
32. Chalmers, S.J.; Bothorel, C.; Clemente, R. Big Data-State of the Art. Available online: **https://hal.archives-ouvertes.fr/hal-00903966** (accessed on 19 January 2019).
33. Apache Software Foundation. Apache Mahout: Scalable Machine Learning and Data Mining. Available online: **https://mahout.apache.org** (accessed on 19 January 2019).
34. Apache Software Foundation. Apache Spark. Available online: **https://spark.apache.org** (accessed on 19 January 2019).
35. Paradigm4. SciDB—The Array Database Management System. Available online: **https://www.paradigm4.com/technology/Open_Source** (accessed on 19 January 2019).
36. Architects Zone. Performance of Graph vs. Relational Databases. Available online: **http://architects.dzone.com/articles/performance-graph-vs** (accessed on 19 January 2019).
37. Malewicz, G.; Austern, M.H.; Bik, A.J.; Dehnert, J.C.; Horn, I.; Leiser, N.; Czajkowski, G. Pregel: A System for Large-scale Graph Processing. In Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, Athens, Greece, 6–10 June 2010; pp. 135–146. [**Google Scholar**]
38. Apache Software Foundation. Giraph—Welcome To Apache Giraph! Available online: **http://giraph.apache.org** (accessed on 19 January 2019).
39. Apache Software Foundation. Hama—The Apache Software Foundation! Available online: **http://hama.apache.org** (accessed on 19 January 2019).
40. Valiant, L.G. A Bridging Model for Parallel Computation. Commun. ACM **1990**, 33, 103–111. [**Google Scholar**] [**CrossRef**]
41. Bisseling, R.H. Parallel Scientific Computation: A Structured Approach Using BSP and MPI, 1st ed.; Oxford University Press: Oxford, UK, 2004. [**Google Scholar**]
42. Curtin, R.R.; Cline, J.R.; Slagle, N.P.; March, W.B.; Ram, P.; Mehta, N.A.; Gray, A.G. MLPACK: A Scalable C++ Machine Learning Library. J. Mach. Learn. Res. **2013**, 14, 801–805. [**Google Scholar**]
43. Low, Y.; Bickson, D.; Gonzalez, J.; Guestrin, C.; Kyrola, A.; Hellerstein, J.M. Distributed GraphLab: A Framework for Machine Learning and Data Mining in the Cloud. Proc. VLDB Endow. **2012**, 5, 716–727. [**Google Scholar**] [**CrossRef**]
44. Parra, V.M.; Syed, A.; Mohammad, A.; Halgamuge, M.N. Pentaho and Jaspersoft: A Comparative Study of Business Intelligence Open Source Tools Processing Big Data to Evaluate Performances. Int. J. Adv. Comput. Sci. Appl. **2016**, 7, 20–29. [**Google Scholar**]
45. NewGenApps. R vs. Python for Data Sciences and Statistics—The Ultimate Comparison. Available online: **https://www.newgenapps.com/blog/r-vs-python-for-data-science-big-data-artificial-intelligence-ml** (accessed on 19 January 2019).
46. Subramanian, V. Deep Learning with PyTorch, 1st ed.; Packt Publishing: Birmingham, UK, 2018. [**Google Scholar**]
47. Chollet, F. Deep Learning with Python, 1st ed.; Manning Publications: Greenwich, CT, USA, 2017. [**Google Scholar**]
48. Miotto, R.; Wang, F.; Wang, S.; Jiang, X.; Dudley, J.T. Deep learning for healthcare: Review, opportunities and challenges. Brief Bioinform. **2018**, 19, 1236–1246. [**Google Scholar**] [**CrossRef**]
49. Krittanawong, C.; Zhang, H.; Wang, Z.; Aydar, M.; Kitai, T. Artificial Intelligence in Precision Cardiovascular Medicine. J. Am. Coll. Cardiol. **2017**, 69, 2657–2664. [**Google Scholar**] [**CrossRef**]
50. Faust, O.; Hagiwara, Y.; Hong, T.J.; Lin, O.S.; Acharya, U.R. Deep Learning for Healthcare Applications Based on Physiological Signals: A Review. Comput. Methods Programs Biomed. **2018**, 161, 1–13. [**Google Scholar**] [**CrossRef**]
51. Murphy KP. Machine learning: a probabilistic perspective. 1st ed. Cambridge: The MIT Press;2012. p. 25.
52. Fenton JJ, Taplin SH, Carney PA, Abraham L, Sickles EA, D'Orsi C, et al. Influence of computer-aided detection on performance of screening mammography. N Engl J Med. 2007; 356:1399–1409.
53. Lehman CD, Wellman RD, Buist DS, Kerlikowske K, Tosteson AN, Miglioretti DL, et al. Diagnostic accura cy of digital screening mammography with and without computer-aided detection. JAMA Intern Med. 201 5; 175:1828–1837
54. Wang Q, Garrity GM, Tiedje JM, Cole JR. Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial t axonomy. Appl Environ Microbiol. 2007; 73:5261–5267.
55. Byvatov E, Fechner U, Sadowski J, Schneider G. Comparison of support vector machine and artificial neural network systems for dr ug/nondrug classification. J Chem Inf Comput Sci. 2003; 43:1882–1889.
56. In : Tong S, Chang E, editors. Support vector machine active learning for image retrieval. Proceedings of the 9th ACM International Conference on Multimedia; 2001 October 5-September 30; Ottawa, Canada. New York: ACM;2001. p. 107.
57. Arbib MA. The handbook of brain theory and neural networks. 2nd ed. Boston: The MIT Press;2003.
58. Drucker H, Burges CJC, Kaufman L, Smola A, Vapnik V. Support vector regression machines. Adv Neural Inf Process Syst. 1997; 9 :155–161.
59. Yu PS, Chen ST, Chang IF. Support vector regression for real-time flood stage forecasting. J Hydrol (Amst). 2006; 328:704–716.
60. Tay FEH, Cao L. Application of support vector machines in financial time series forecasting. Omega (Westport). 2001; 29:309–317.
61. Deep Learning in Medical Imaging: General Overview, Korean Journal of Radiology 2017; 18(4): 570-584., June-Goo Lee, Sanghoo n Jun, Young-Won Cho, Hyunna Lee, Guk Bae Kim, Joon Beom Seo, Namkug Kim