International Journal of Trend in Research and Development, Volume 6(5), ISSN: 2394-9333
www.ijtrd.com

Review of Some Machine Learning Techniques for Big Data, Their Uses, Strengths and Weaknesses

[1]Asogwa D.C, [2]Anigbogu S.O, [3]Onyesolu M.O and [4]Chukwuneke C.I,
[1,2,3,4]Department of Computer Science, Faculty of Physical Sciences, NnamdiAzikiwe University, Awka, Nigeria

Abstract: Machine learning is a field of computer science which gives computers an ability to learn without being explicitly programmed. Machine learning is used in a variety of computational tasks where designing and programming explicit algorithms with good performance is not easy.Big data are now rapidly expanding in all science and engineering domains. The potentials of these increased volumes of data are obviously very significant to every aspect of our lives. To aid us in decision making and future predictions requires new ways of thinking and new learning techniques to address the various challenges.Traditional analytical approaches are insufficient to analyze big data because they are highly scalable and unstructured data captured in real time. Machine learning (ML), addresses this challenge, by enabling a system to automatically learn patterns from data that can be leveraged in future predictions.This paper reviews some machine learning techniques for big data highlighting their uses/applications, strengths and weaknesses in learning data patterns. The techniques reviewed include Bayesian network, association rules, naïve bayes, decision trees, nearest neighbor and super vector machines (SVM).

Keywords: *Machine Learning, Supervised Learning, Unsupervised Learning, Classification, Big Data*

I. INTRODUCTION

Machine learning is a branch of artificial intelligence that allows computer systems to learn directly from examples, data, and experience. Through enabling computers to perform specific tasks intelligently, machine learning systems can carry out complex processes by learning from data, rather than following pre-programmed rules.Big Data generally refers to data that exceeds the typical storage, processing, and computing capacity of conventional databases and data analysis techniques. As a resource, Big Data requires tools and methods that can be applied to analyze and extract patterns from large-scale data. The rise of Big Data has been caused by increased data storage capabilities, increased computational processing power, and availability of increased volumes of data, which give organization more data than they have computing resources and technologies to process. In addition to the obvious great volumes of data, Big Data is also associated with other specific complexities, often referred to as the four Vs: Volume, Variety, Velocity, and Veracity (Grolinger, et al 2014). The unmanageable large Volume of data poses an immediate challenge to conventional computing environments and requires scalable storage and a distributed strategy to data querying and analysis. However, this large Volume of data is also a major positive feature of Big Data. Many companies, such as Facebook, Yahoo, Google, already have large amounts of data and have recently begun tapping into its benefits (Almeida & Bernardino, 2015). A general theme in Big Data systems is that the raw data is increasingly diverse and complex, consisting of largely un-categorized/unsupervised data along with perhaps a small quantity of categorized/ supervised data. Working with the Variety among different data representations in a given repository poses unique challenges with Big Data, which requires Big Data preprocessing of unstructured data in order to extract structured/ordered representations of the data for human and/or downstream consumption. In today's data-intensive technology era, data Velocity – the increasing rate at which data is collected and obtained – is just as important as the Volume and Variety characteristics of Big Data. While the possibility of data loss exists with streaming data if it is generally not immediately processed and analyzed, there is the option to save fast-moving data into bulk storage for batch processing at a later time. However, the practical importance of dealing with Velocity associated with Big Data is the quickness of the feedback loop, that is, process of translating data input into useable information. This is especially important in the case of time-sensitive information processing. Some companies such as Twitter, Yahoo, and IBM have developed algorithms that address the analysis of streaming data (Wu et al, 2014). Veracity in Big Data deals with the trustworthiness or usefulness of results obtained from data analysis, and brings to light the old adage "Garbage-In-Garbage-Out" for decision making based on Big Data Analytics. As the number of data sources and types increases, sustaining trust in Big Data Analytics presents a practical challenge.

Algorithm models for dealing with big data take different shapes, depending on their purpose. Using different algorithms to provide comparisons can offer some surprising results about the data being used. They can come as a collection of scenarios, an advanced mathematical analysis, or even a decision tree. Some models function best only for certain data and analyses. For example, classification algorithms with decision rules can be used to screen out problems, such as a loan applicant with a high probability of defaulting.

Unsupervised clustering algorithms can be used to find relationships within an organization's dataset. These algorithms can be used to find different kinds of groupings within a customer base, or to decide what customers and services can be grouped together. An unsupervised clustering approach can offer some distinct advantages, as compared to the supervised learning approaches. One example is the way novel applications can be discovered by studying how the connections are grouped when a new cluster is formed.

They are different existing models for machine learning on big data and they include: Decision Tree based model, linear regression based model, Neural Network, Bayesian Network, Nearest Neighbor and many others.

Brief review of some machine learning techniques

Generally, the field of machine learning is divided into three subdomains: supervised learning, unsupervised learning, and reinforcement learning (Adams et al, 2008). Briefly, supervised learning requires training with labeled data which has inputs and desired outputs. In contrast with the supervised

learning, unsupervised learning does not require labeled training data and the environment only provides inputs without desired targets. Reinforcement learning enables learning from feedback received through interactions with an external environment. Based on these three essential learning paradigms, a lot of theory mechanisms and application services have been proposed for dealing with data tasks.

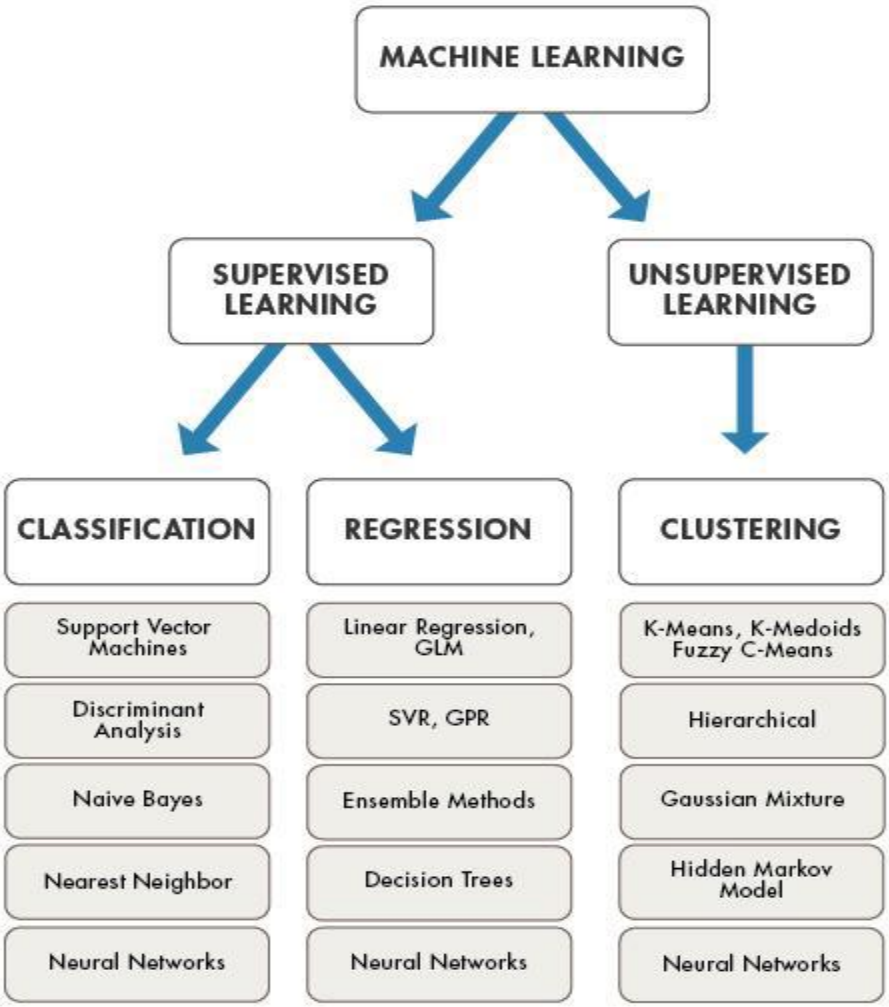

Fig 1: machine learning algorithms (Diksha Sharma &Neeraj Kumar, 2017)

Bayesian Network

A Bayesian network is a graphical model that consists of two parts, *G*, *P*, where *G* is a directed acyclic graph (DAG) whose nodes represent random variables and arcs between nodes represent conditional dependency of the random variables and *P* is a set of conditional probability distributions, one for each node conditional on its parents. The conditional probability distributions for each node may be prior or learned. When building Bayesian networks from prior knowledge alone, the probabilities will be prior (Bayesian). When learning the networks from data, the probabilities will be posterior (learned).

Bayesian networks do not necessarily imply that they rely on Bayesian statistics. Rather, they are called because they use Bayes' rule for probabilistic inference. It is possible to use Bayesian statistics to learn a Bayesian network, but there are also many other techniques that are more closely related to traditional statistical methods. For example, it is common to use frequents methods to estimate the parameters of the conditional probability distributions. Based on the topology of the structure, there are different types of networks.

Bayesian networks can be used for both supervised learning and unsupervised learning. In unsupervised learning, there is no target variable; therefore, the only network structure is directed acyclic graph (DAG). In supervised learning, we need only the variables that are around the target, that is, the parents, the children, and the other parents of the children (spouses). Besides the naive Bayes, there are other types of network structures:

i Tree augmented naive Bayes (TAN)
ii Bayesian network augmented Naive Bayes (BAN)
iii Parent child Bayesian network (PC) and
iv Markov blanket Bayesian network (MB).

These network structures differ in which links are allowed between nodes. They are classified based on the three types of links (from target to input, from input to target, and between the input nodes) and whether to allow spouses of the target.

Strengths of Bayesian network:

It is straight forward to create a network, create the nodes and connect them and then assign probabilities and conditional probabilities. When existing observations are applied, the overall results change due to the nature of the graph.

Weaknesses:

Coding can be very difficult and require some proper planning on paper.

To make the final prediction output more accurate, a domain expert is needed to help with the initial values of the probabilities.

Naives Bayes

Naïve Bayes gives a simple approach, with clear semantics, to representing, using, and learning probabilistic knowledge. Impressive results can be achieved using it. It has often been shown that Naïve Bayes rivals and indeed outperforms more sophisticated classifiers on many datasets. The moral is, always try the simple things first. Repeatedly in machine learning people have eventually, after an extended struggle, obtained good results using sophisticated learning methods only to discover years later that simple methods such as 1R and Naïve Bayes do just as well—or even better.

There are many datasets for which Naïve Bayes does not do so well, however, and it is easy to see why. Because attributes are treated as though they were completely independent, the addition of redundant ones skews the learning process. Dependenciesbetween attributes inevitably reduce the power of Naïve Bayes to discern what is going on. They can, however, be ameliorated by using a subset of the attributes in the decision procedure, making a careful selection of which ones to use.

Strengths:

Naïve Bayes gives a simple approach with clear semantics, to representing, using and learning probabilistic knowledge. It is often shown that it outperforms more sophisticated classifiers on many datasets. Impressive results can be achieved using it.

Weaknesses:

The addition of redundant attributes skews the learning process because they are treated as though they were completely independent.

Super Vector Machines (SVM)

SVM offers a principled approach to machine learning problems because of its mathematical foundation in statistical learning theory. SVM constructs its solution in terms of a subset of the training input. SVM has been extensively used for classification, regression, novelty detection tasks, and feature reduction.

In classification tasks a discriminant machine learning technique aims at finding, based on an independent and identically distributed (iid) training dataset, a discriminant function that can correctly predict labels for newly acquired instances. Unlike generative machine learning approaches, which require computations of conditional probability distributions, a discriminant classification function takes a data point *x* and assigns it to one of the different classes that are a part of the classification task. Less powerful than generative approaches, which are mostly used when prediction involves outlier detection, discriminant approaches require fewer computational resources and less training data, especially for a multidimensional feature space and when only posterior probabilities are needed. From a geometric perspective, learning a classifier is equivalent to finding the equation for a multidimensional surface that best separates the different classes in the feature space.

SVM is a discriminant technique, and, because it solves the convex optimization problem analytically, it always returns the same optimal hyperplane parameter—in contrast to genetic algorithms (GAs) or perceptron, both of which are widely used for classification in machine learning. For perceptron, solutions are highly dependent on the initialization and termination criteria.For a specific kernel that transforms the data from the input space to the feature space, training returns uniquely defined SVM model parameters for a given training set, whereas the perceptron and GA classifier models are different each time training is initialized. The aim of GAs and perceptron is only to minimize error during training, which will translate into several hyperplanes' meeting this requirement. If many hyperplanes can be learned during the training phase, only the optimal one is retained, because training is practically performed on samples of the population even though the test data may not exhibit the same distribution as the training set. When trained with data that are not representative of the overall data population, hyperplanes are prone to poor generalization.

Strengths:

- SVM's are very good when one has no idea on the data.
- Works well with even unstructured and semi structured data like text, Images and trees.
- The kernel trick is real strength of SVM. With an appropriate kernel function, one can solve any complex problem.
- Unlike in neural networks, SVM is not solved for local optima.
- It scales relatively well to high dimensional data.
- SVM models have generalization in practice; the risk of overfitting is less in SVM.

Weaknesses

- Choosing a "good" kernel function is not easy.
- Long training time for large datasets.
- Difficult to understand and interpret the final model, variable weights and individual impact.
- Since the final model is not so easy to see, one cannot do small calibrations to the model hence it is tough to incorporate business logic.

Association rules:

Association rules are often sought for very large datasets, and efficient algorithms are highly valued. In practice, the amount of computation needed to generate association rulesdepends critically on the minimum coverage specified. The accuracy has lessinfluence because it does not affect the number of passes that one must makethrough the dataset. In many situations one will want to obtain a certain numberof rules—say 50—with the greatest possible coverage at a prespecified minimum accuracy level. One way to do this is to begin by specifying the coverageto be rather high and to then successively reduce it, re-executing the entirerule-finding algorithm for each coverage value and repeating this until thedesired number of rules has been generated.

Association rules are often used when attributes are binary—eitherpresent or absent—and most of the attribute values associated with a giveninstance are absent.

Uses/applications:

- Market Basket Analysis: given a database of customer transactions, where each transaction is a set of items the goal is to find groups of items which are frequently purchased together.

- Telecommunication (each customer is a transaction containing the set of phone calls)
- Credit Cards/ Banking Services (each card/account is a transaction containing the set of customer's payments)
- Medical Treatments (each patient is represented as a transaction containing the ordered set of diseases)
- Basketball-Game Analysis (each game is represented as a transaction containing the ordered set of ball passes)

Strengths:

- Uses large itemset property.
- Easily parallelized
- Easy to implement.

Weaknesses:

- Assumes transaction database is memory resident.
- Requires many database scans.

Nearest neighbor:

Nearest-neighbor instance-based learning is simple and often works very well.*k*-nearest-neighbor strategy is adopted, where some fixed,small, number *k* of nearest neighbors—say five—are located and used togetherto determine the class of the test instance through a simple majority vote. Another way of proofing the database against noise is to choosethe exemplars that are added to it selectively and judiciously.

The nearest-neighbor method originated many decades ago, and statisticiansanalyzed *k*-nearest-neighbor schemes in the early 1950s. If the number of traininginstances is large, it makes intuitive sense to use more than one nearestneighbor, but clearly this is dangerous if there are few instances. It can be shownthat when *k* and the number *n* of instances both become infinite in such a waythat *k*/*n* A0, the probability of error approaches the theoretical minimum forthe dataset. The nearest-neighbor method was adopted as a classificationmethod in the early 1960s and has been widely used in the field of pattern recognitionfor more than three decades.

Nearest-neighbor classification was notoriously slow until *k*D-trees began tobe applied in the early 1990s, although the data structure itself was developedmuch earlier. In practice, these trees become inefficient when the dimension ofthe space increases and are only worthwhile when the number of attributes issmall—up to 10. Ball trees were developed much more recently and are aninstance of a more general structure sometimes called a metric tree. Sophisticatedalgorithms can create metric trees that deal successfully with thousandsof dimensions.Instead of storing all training instances, one can compress them into regions.Numeric attributes can be discretized intointervals, and "intervals" consisting of a single point can be used for nominalones. Then, given a test instance, one can determine which intervals it residesin and classify it by voting, a method called voting feature intervals. Thesemethods are very approximate, but very fast, and can be useful for initial analysisof large datasets.

Strengths:

- Simple technique that is easily implemented
- Building model is cheap
- Extremely flexible classification scheme
- Well suited forMulti-modal classes, records with multiple class labels
- Error rate at most twice that of Bayes error rate. (Michiro et al, 2005)

Weaknesses:

- Classifying unknown records are relatively expensive
- Requires distance computation of k-nearest neighbors
- Computationally intensive, especially when the size of the training set grows
- Accuracy can be severely degraded by the presence of noisy or irrelevant features

Decision trees

Decision tree learning is a method for approximating discrete-valued target functions, in which the learned function is represented by a decision tree. Learned trees can also be re-represented as sets of if-then rules to improve human readability. These learning methods are among the most popular of inductive inference algorithms and have been successfully applied to a broad range of tasks. In machine learning, decision trees partition the data set in appropriate values until a tree structure has emerged. This process is called recursive partitioning (Strobl, 2009).

Decision tree algorithm tries to find the best way to partition the data so that parts are as homogeneous as possible. If a fully homogeneous part is impossible, more common value is chosen.

The aim with any decision tree is to create a workable model that will predictthe value of a target variable based on the set of input variables.

Uses for Decision Trees

This includes how one selects different options within an automated telephonecall. The options are essentially decisions that are being made for one to getto the desired department. These decision trees are used effectively in manyindustry areas.

Financial institutions use decision trees. One of the fundamental use cases isin option pricing, where a binary-like decision tree is used to predict the priceof an option in either a bull or bear market.Marketers use decision trees to establish customers by type and predictwhether a customer will buy a specific type of product.In the medical field, decision tree models have been designed to diagnoseblood infections or even predict heart attack outcomes in chest pain patients.Variables in the decision tree include diagnosis, treatment, and patient data.The gaming industry now uses multiple decision trees in movement recognitionand facial recognition. The Microsoft Kinect platform uses this method totrack body movement. The Kinect team used one million images and trainedthree trees. Within one day, and using a 1,000-core cluster, the decision trees were classifying specific body parts across the screen.

Strengths of Decision Trees

They are easyto read. After a model is generated, it is easy to report back to others regardinghow the tree works. Also, with decision trees one can handle numericalor categorized information.

In terms of data preparation, there is not much to do. As long as the data is formalizedin something like comma separated variables, then one can create a workingmodel. This also makes it easy to validate the model using various tests. Withdecision trees one use white-box testing—meaning the

internal workings can be observed but not changed; one can view the steps that are being used whenthe tree is being modeled.Decision trees perform well with reasonable amounts of computing power. The decision tree learning can handle a large set of data well.

Limitations of Decision Trees

One of the main issues of decision trees is that they can createoverly complex models, depending on the data presented in the training set.

To avoid the machine learning algorithm's over-fitting the data, it is sometimesworth reviewing the training data and pruning the values to categories, whichwill produce a more refined and better-tuned model.Some of the decision tree concepts can be hard to learn because the model cannotexpress them easily. This shortcoming sometimes results in a larger-than-normal model. One might be required to change the model or look at different methodsof machine learning.

CONCLUSION

Machine learning algorithms are widely used in variety of applications like digital image processing(image recognition),(Kumar & Gupta, 2016), big data analysis, (Sharma et al, 2014), Speech Recognition, Medical Diagnosis, Statistical Arbitrage, Learning Associations, Classification, Prediction etc.

Each technique has different application areas and is useful in different domains based on its advantages. Thus, keeping in mind the limitations of each of the techniques and also the prime focus being the improvement in performance and efficiency one should use that technique, which best suits a particular application.The article illustrates the concept of machine learning with its uses/applications, strengths and weaknesses.It also highlights the various types of learning such as supervised learning, unsupervised learning and reinforcement learning.

References

[1] B Adam, IFC Smith, &F Asce, 2008, " Reinforcement learning for structural control". J Comput Civil Eng **22**(2), 133–139.

[2] CarolinStrobl, James Malley, and Gerhard Tutz, 2009, "An Introduction to Recursive Partitioning" NCBI, Psychol Methods. 2009 Dec; 14(4): 323–348. doi: 10.1037/a0016973

[3] Diksha Sharma &Neeraj Kumar, 2017, A Review on Machine Learning Algorithms, Tasks and Applications"" International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 6, Issue 10, October 2017, ISSN: 2278 – 1323

[4] Grolinger, M. Hayes, W. Higashino, A. L'Heureux,& D. S. Allison, 2014."Challenges for MapReduce in Big Data", Proc. of the IEEE 10th 2014 World Congress on Services. https://www.researchgate.net/.../263694670_Challenges_for_MapReduce_in_Big_Data

[5] Kumar,N. and Gupta, S., 2016. Offline Handwritten Gurmukhi Character Recognition: A Review. International Journal of Software Engineering and Its Applications, 10(5), pp.77-86.

[6] MichihiroKuramochi and George Karypis, 2005, "Gene Classification using Expression Profiles: A Feasibility Study", International Journal on Artificial Intelligence Tools. Vol. 14, No. 4, pp. 641-660.

[7] Muhammad, I. and Yan, Z., 2015. Supervised Machine Learning Approaches: A Survey. ICTACT Journal on Soft Computing, 5(3).

[8] Pedro Daniel Coimbra de Almeida& Jorge Bernardino, 2015 "Big data open source platforms" Published in: Big Data (BigData Congress), 2015 IEEE International Congress https://ieeexplore.ieee.org/document/7207229/

[9] Sharma, D., Pabby, G. and Kumar, N.,2017, "Challenges Involved in Big Data Processing & Methods to Solve Big Data Processing Problems".IJRASET,5(8),pp.841-844.

[10] Singh, S., Kumar, N. and Kaur, N., 2014. Design and development Of Rfid Based Intelligent Security System. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume, 3.

[11] Talwar, A. and Kumar, Y., 2013. Machine Learning: An artificial intelligence methodology. International Journal of Engineering and Computer Science, 2, pp.3400-3404.

[12] Wu, X., Zhu, X., Wu, G., & Ding, W. (2014). "Data Mining with Big Data". TKDE, 26 (1), 97- 107.