International Journal of Mathematical, Engineering and Management Sciences
Vol. 1, No. 2, 52–61, 2016
ISSN: 2455-7749

Machine Learning in Big Data

Lidong Wang[1,*], Cheryl Ann Alexander[2]

[1]Department of Engineering Technology, Mississippi Valley State University, USA
[2]Technology and Healthcare Solutions, Inc., USA
**Corresponding author*: lwang22@students.tntech.edu

(Received March 15, 2016; Accepted April 2, 2016)

Abstract
Machine learning is an artificial intelligence method of discovering knowledge for making intelligent decisions. Big Data has great impacts on scientific discoveries and value creation. This paper introduces methods in machine learning, main technologies in Big Data, and some applications of machine learning in Big Data. Challenges of machine learning applications in Big Data are discussed. Some new methods and technology progress of machine learning in Big Data are also presented.

Keywords-big data, machine learning, big data analytics, information technology, stream processing.

1. Introduction

Machine learning is an important area of artificial intelligence. The objective of machine learning is to discover knowledge and make intelligent decisions. Machine learning algorithms can be categorized into supervised, unsupervised, and semi-supervised. When big data is concerned, it is necessary to scale up machine learning algorithms (Chen and Zhang, 2014; Tarwani et al., 2015). Another categorization of machine learning according to the output of a machine learning system includes classification, regression, clustering, and density estimation, etc. Machine learning approaches include decision tree learning, association rule learning, artificial neural networks, support vector machines (SVM), clustering, Bayesian networks, and genetic algorithms, etc. (https://en.wikipedia.org/wiki/Machine_learning).

Examples of supervised learning algorithms include Naïve Bayes, boosting algorithm, support vector machines (SVM), and maximum entropy method (MaxENT), etc. Unsupervised learning takes unlabelled data and classifies by comparing data features. Examples of unsupervised learning algorithms include clustering (*k*-means, density-based, and hierarchical, etc.), self-organizing maps (SOM), and adaptive resonance theory (ART) (Jaswant and Kumar, 2015).

Machine learning has been used in big data. Big data is a massive volume of both structured and unstructured data that is so large that it is difficult to process using traditional database and software techniques. Big data technologies have great impacts on scientific discoveries and value creation (Demchenko et al., 2013; O'Leary, 2013; Jagadish et al., 2014). Massive parallel-processing (MPP), distributed file systems, and cloud computing, etc. support Big Data (Zaslavsky et al., 2012). Besides general cloud infrastructure services, technologies such as Hadoop, Databases/Servers SQL, NoSQL, and MPP databases, etc. are also used to support Big Data (Turk, 2012).

This paper introduces machine learning, its applications in Big Data, and the challenges and technology progress of machine learning in Big Data. The organization of this paper is as follows: the next section introduces methods of machine learning and big data; Section 3 introduces machine learning applications in big data; Section 4 discusses challenges of machine learning applications in big data; Section 5 presents technology progress of machine learning applications in big data; and the final section is conclusions.

2. Methods of Machine Learning and Big Data

Supervised learning can be categorized into classification and regression. When the class attribute is discrete, it is classification; when the class attribute is continuous, it is called regression. Decision tree learning, naive Bayes classifier, and k-nearest neighbor (k-NN), etc. are classification methods. Linear regression and logistic regression are regression methods. Unsupervised learning divides instances into groups of similar objects (Zafarani et al., 2014).

Clustering can be grouped into three categories. They are supervised, unsupervised, and semi-supervised (Dean, 2014):

- Supervised clustering: It identifies clusters that have high probability densities with respect to individual classes. It is used when there is a target variable and a training set including the variables to cluster.
- Unsupervised clustering: It maximizes the intracluster similarity and minimizes the intercluster similarity when a similarity/dissimilarity measure is given. It uses a specific objective function. *K*-means and hierarchical clustering are the most commonly used unsupervised clustering methods in segmentation.
- Semi-supervised clustering: In addition to the similarity measure, semi-supervised clustering uses other guiding/adjusting domain information to improve clustering. This domain information can be pairwise constraints between the observations or target variables for some of the observations.

Decision trees classify data based on their feature values. Decision trees are constructed recursively from training data using a top-down greedy approach in which features are sequentially selected (Zafarani et al., 2014). Decision tree classifiers organize the training data into a tree-structure plan. Decision trees are constructed by stating with the root node having the whole data set, iteratively choosing splitting criteria and expanding leaf nodes with partitioned data subsets according to the splitting criteria. Splitting criteria are chosen based on some quality measures such as information gain, which requires handling the entire data set of each expanding nodes. This makes it difficult for decision trees to be applied to big data applications (Lee, 2014).

Support vector machine (SVM) is a binary classifier which finds linear classifier in higher dimensional feature space to which original data space is mapped. SVM shows very good performance for data sets in a moderate size. It has inherent limitations to big data applications (Lee, 2014).

Deep machine learning has become a research frontier in artificial intelligence. It is a machine learning technique, where many layers of information processing stages are exploited in hierarchical architectures. It computes hierarchical features or representations of the observational data, where the higher-level features are defined from lower-level ones. Deep learning algorithms extract high-level, complex abstractions as data representations through a hierarchical learning process. While deep learning can be applied to learn from labeled data, it is primarily attractive for learning from large amounts of unlabeled data, making it attractive to extract meaningful representations or patterns from big data. Deep learning algorithms and architectures are more aptly suited to address issues related to Volume and Variety of Big data analytics. Deep machine learning can be applied to big data. However, it has some restrictions in big data applications because it requires significant amount of training time (Lee, 2014; Najafabadi et al., 2015).

Parallel learner for assembling numerous ensemble trees (PLANET) is a regression tree algorithm implemented with a sequence of MapReduce jobs that run on the big data framework, Hadoop. It can deal with big volume of data, but is not applicable to data with categorical attributes (Lee, 2014).

One trend in machine learning is to combine results of multiple learners to obtain better accuracy. This trend is commonly known as Ensemble Learning. There are four methods of combining multiple models: bagging, boosting, stacking, and error-correcting output (Wang et al., 2014).

A comparison of several machine learning algorithms was made in Table 1 (Li, 2015) according to algorithms type, algorithms trait, learning policy, learning algorithms, and classification strategy. Some features of machine learning algorithms were compared in Table 2 (Kotsiantis, 2007).

Algorithms	**Algorithms type**	**Algorithms characteristic**	**Learning policy**	**Learning algorithms**	**Classification strategy**
Decision tree	Discriminant	Classification tree	Regularized maximum likelihood estimation	Feature selection, generation, prune	IF-THEN rule According to tree spitting
Non-linear SVM (based on libsvm)	Discriminant	Super-plane separation, kernel trick	Minimizing regular hinge loss, soft margin maximization	Sequential minimal optimization algorithm (SMO)	Maximum class of test samples
Linear SVM (based on liblinear)	Discriminant	Super-plane separation	Minimizing the loss of regular hinge, soft margin maximization	Sequential dual method	Maximum weighted test sample
Stochastic gradient boosting	Discriminant	Linear combination of weak classifier (based on decision tree)	Addition minimization loss	Stochastic gradient descent algorithm	Linear combination of weighted maximum weak classifiers
Naive Bayesian classifier	Generative	Joint distribution of class and feature, conditional independent assumption	Estimation of maximum likelihood, Maximum posterior probability	Probabilistic computation	Maximum posterior probability

Table 1. Summary of several machine learning algorithms

There are several frameworks, like Map/Reduce, DryadLINQ, and IBM parallel machine learning toolbox that have capabilities to scale up machine learning (Chen and Zhang, 2014). Mahout is an open source machine learning library from Apache for big data analysis. It aims to be the machine learning tool of choice when the collection of data is very large (Jaswant and Kumar, 2015). The Apache Mahout project aims at building a scalable machine learning library on top of Hadoop. The Mahout machine learning library was integrated, adapted, and extended by developing advanced machine learning algorithms for large scale data. Mahout may greatly help towards grouping similar items, identifying main topics, assigning items to predefined categories,

	Decision Trees	**Neural Networks**	**Naïve Bayes**	**kNN**	**SVM**	**Rule-learners**
Accuracy in general	..	...	.	..	....	..
Learning speed with respect to attributes number and instances number	...	.	....	....	.	..
Classification speed	....	....	....	.	....	....
Tolerance to missing values	...	.	....	.	..	..
Tolerance to irrelevant attributes	...	.	..	..	....	..
Tolerance to redundant attributes	..	..	.	..	....	..
Tolerance to highly interdependent attributes (e.g. parity problems)	..	...	.	.	...	..
Dealing with discrete/binary/continuous attributes	....	... (not discrete)	... (not continuous)	... (not directly discrete)	.. (not discrete)	... (not directly continuous)
Tolerance to noise	..	..	...	.	..	.
Handling the danger of overfitting	..	.	...	...	..	..
Attempts for incremental learning	..	...	....	....	..	.
Ability of explanation /transparency of knowledge/classifications	....	.	....	..	.	....
Dealing with model parameters	...	.	....	...	.	...

Table 2. Comparison of machine learning algorithms
(.... dots represent the best performance; . dot represents the worst performance)

recommending significant data to diverse stakeholders, and discovering frequent and meaningful patterns in a specific decision-making setting (Karacapilidis et al., 2013).

PivotalR is a package of machine learning in big data. PivotalR uses the full power of parallel computation and distributive storage, and therefore gives a normal *R* user access to big data stored in distributive databases or Hadoop distributive file system (HDFS). It provides data-parallel implementations of mathematical, statistical and machine-learning algorithms for structured and unstructured data. Therefore, PivotalR also enables a user to apply machine learning algorithms on big data (Qian, 2014).

There are a lot of technologies supporting Big Data analytics and applications. Table 3 (Dean, 2014) compares a number of big data technologies. The table highlights the different types of systems and their comparative strengths and weaknesses.

	In-memory database	**MPP database**	**Big Data appliance**	**Hadoop**	**NoSQL database**
Consistent	W	W	W	P	P
Available	W	W	W	P	P
Fault tolerant	W	W	P	W	W
Suitable for real-time transactions	W	W	W	F	F
Suitable for analytics	P	P	W	W	F
Suitable for extremely big data	F	P	P	W	W
Suitable for unstructured data	F	F	P	W	W

W: Meets widely held expectations.
P: Potentially meets widely held expectations.
F: Fails to meet widely held expectations.

Table 3. Comparison of Big Data Technologies

3. Examples of Machine Learning Applications in Big Data

The combination of supervised and unsupervised machine learning techniques for efficiently analyzing a big volume of crime data was proposed. The combination includes three steps: dimensionality reduction, clustering, and classification. *R* statistical software was used because it is a powerful tool to deal with big data. The specific work is outlined as follows (Nasridinov, 2014):

- Measure correlation between crime and social attributes. This method reduces dimensionality of the crime data.
- Use unsupervised machine learning technique to divide crime data into groups; use *k*-means clustering algorithm to cluster the crime data into dangerous, average, and safe regions.
- Use supervised machine learning technique to predict whether a particular region is dangerous or safe; use decision tree classification algorithm to perform predictions.

Analysis and mining of social network data for society issues was conducted using Big Data. Social data mining is the process of analyzing, representing as well as extracting actionable patterns from social network data. Machine learning and stemming algorithms were used to classify the tweets. Tweets are often in the pattern of big data. The predicting features from tweets were extracted from a collection of tweets; stopping words were removed; and all keywords were selected. As tweets are very short and may contain incomplete sentences, the meaning of the tweets may be ambiguous. In machine learning, support vector machines (SVM) are supervised models with related learning algorithms that analyze all the data which are used for classification of the tweets. Stemming algorithm uses a pre-processing task in text mining and can be used as a common requirement of natural language processing functions. Stemming algorithm was used to extract the main keywords or root words from the tweets. The stemming algorithm can be applied to predict the keywords from the tweets. All the keywords are classified by the SVM algorithm (Kanagavalli et al., 2015).

4. Challenges of Machine Learning Applications in Big Data

General challenges about machine learning are: (i) designing scalable and flexible computational architectures for machine learning; (ii) the ability to understand characteristics of data before applying machine learning algorithms and tools; and (iii) the ability to construct, learn and infer with increasing sample size, dimensionality, and categories of labels (Sukumar, 2014).

There are many scale machine learning algorithms, but many important specific sub-fields in large-scale machine learning, such as large-scale recommender systems, natural language processing, association rule learning, ensemble learning, still face the scalability problems (Chen and Zhang, 2014).

The basic MapReduce framework commonly provided by first-generation "Big Data analytics" platforms like Hadoop lacks an essential feature for machine learning. MapReduce does not support iteration /recursion or certain key features required to efficiently iterate "around" a MapReduce program. Programmers building machine learning models on such systems have to implement looping in ad-hoc ways outside the core MapReduce framework. This lack of support has motivated the recent development of various specialized methods or libraries to support iterative programming on large clusters. Meanwhile, recent MapReduce extensions such as HaLoop, Twister, and PrItr aim at directly addressing the iteration outage in MapReduce (Bu et al., 2012).

Major problems that make the machine learning (ML) methods unsuitable for solving big data classification problems are: (i) An ML method that is trained on a particular labeled datasets may not be suitable for another dataset – that the classification may not be robust over different datasets; (ii) an ML method is generally trained using a certain number of class types and thus a large varieties of class types found in a dynamically growing dataset will lead to inaccurate classification results; and (iii) an ML method is developed based on a single learning task, and therefore they are not suitable for today's multiple learning tasks and knowledge transfer requirements of Big data analytics (Suthaharan, 2014).

Traditional algorithms in ML generally do not scale to big data. The main difficulty lies with their memory constraint. Although algorithms typically assume that training data samples exist in main memory, big data does not fit into it. A common method of learning from a large dataset is data distribution. By replacing batch training on the original training dataset with separated computations on the distributed subsets, one can train an alternative prediction model at a sacrifice of accuracy. Another approach is using online learning, in which memory usage does not depend on dataset size. Both online learning and distributed learning are not sufficient for learning from big data streams. There are two reasons. First is that the data size is too big to be relaxed by either online or distributed learning. Sequential online learning on big data requires too much time for training on a single machine. On the other hand, distributed learning with a big number of machines reduces the gained efficiency per machine and affects the overall performance. The second reason is that combining real-time training and prediction has not been studied. Big data is used after being stored in (distributed) storage; therefore, the learning process also tends to work in a batch manner (Hido et al., 2013).

Scaling up big data to proper dimensionality is a challenge that can encounter in machine learning algorithms; and there are challenges of dealing with velocity, volume and many more for all types of machine learning algorithms. Since big data processing requires decomposition, parallelism, modularity and/or recurrence, inflexible black-box type machine learning models failed in an outset (Tarwani et al., 2015).

Applying the distributed data-parallelism (DDP) patterns in Big Data Bayesian Network (BN) learning faces several challenges: (i) effectively pre-processing big data to evaluate its quality and reduce the size if necessary; (ii) designing a workflow capable of taking Gigabytes of big data sets and learning BNs with decent accuracy; (iii) providing easy scalability support to BN learning algorithms (Wang et al., 2014).

Deep learning challenges in big data analytics lie in: incremental learning for non-stationary data, high-dimensional data, and large-scale models (Najafabadi et al., 2015). Because high-level data parallel frameworks, like MapReduce do not naturally or efficiently support many important data mining and machine learning algorithms and can lead to inefficient learning systems, the GraphLab abstraction was introduced. It naturally expresses asynchronous, dynamic, graph-parallel computation while ensuring data consistency and achieving a high degree of parallel performance in the shared-memory setting (Low et al., 2012).

5. Technology Progress of Machine Learning Applications in Big Data

Most advances for scalable machine learning (e.g. Madlib, Apache Mahout, etc.) happen in the massively parallel database processing community. Better work can be done in the Big Data era by designing and implementing ML algorithms with scale-friendly predictive functions. The following methods have been exploring and evaluating (Sukumar, 2014): (i) deep learning algorithms that automate the feature engineering process by learning to create and sift through data-driven features, (ii) incremental learning algorithms in associative memory architectures that can seamlessly adapt to future datasets and sources, (iii) faceted learning that can learn hierarchical structure in the data, and (iv) multi-task learning that can learn a number of predictive functions in parallel.

Big Data classification requires multi-domain, representation-learning (MDRL) method because of its large and growing data domain. The MDRL method includes feature variable learning, feature extraction learning, and distance-metric learning. Several representation-learning methods have been proposed in ML. The cross-domain, representation-learning (CDRL) method is maybe suitable for the Big Data classification along with the suggested network model (Suthaharan, 2014).

A significant benefit of deep learning is the analysis and learning of large amounts of unsupervised data, making it a very useful tool for Big Data analytics. How deep learning can be used in Big Data analytics was explored; this includes extracting complex patterns from big volumes of data, semantic indexing, data tagging, fast information retrieval, and simplifying discriminative tasks. Some further research of deep learning in Big Data was also investigated; this includes streaming data, high-dimensional data, scalability of Deep Learning models, and distributed computing (Najafabadi et al., 2015).

As an important machine learning technique, Bayesian Network (BN) has been widely used to model probabilistic relationships among variables. An intelligent Big Data pre-processing approach and a data quality score were proposed to measure and ensure the data quality and data faithfulness; a new weight based ensemble algorithm was proposed to learn a BN structure from an ensemble of local results. For easily integrating the algorithm with distributed data-parallelism (DDP) engines, such as Hadoop, Kepler scientific workflow was employed to build the whole learning process. How Kepler can facilitate building and running the Big Data BN learning application was also demonstrated. A Scalable Bayesian Network Learning (SBNL) workflow was designed through combining machine learning, distributed computing, and workflow techniques. The workflow includes intelligent Big Data pre-processing and effective BN learning from Big Data by leveraging ensemble learning and distributed computing model (Wang et al., 2014).

For stream processing, one must process new data in real-time and in a lot of times, considers historical data as well to generate a value. Most often, stream processing involves using previously trained models to avoid too much processing and finally reduce response times. A novel architecture to perform ML over big data streams was proposed. The architecture provides reliable persistent storage of data over the Hadoop Distributed File System (HDFS) and HBase. The core of the architecture includes the batch- and stream-processing modules. It provides ML tools and algorithms so that developers can easily take advantage of them to carry out tasks such as prediction, clustering, recommendation, and classification, etc. (Baldominos et al., 2014).

A distributed streaming algorithm was developed to learn decision rules for regression tasks. The algorithm is available in Scalable Advanced Massive Online Analysis (SAMOA), an open-source platform for mining big data streams. It uses a hybrid of vertical and horizontal parallelism to distribute Adaptive Model Rules (AMRules) on a cluster. The decision rules built by AMRules are comprehensible models. SAMOA eases the development of new distributed ML algorithms and the deployment of these implementations on top of state-of the-art distributed stream processing engines (DSPEs). It is also a library of distributed ML algorithms allowing users to use or customize existing ones (Vu et al., 2014).

Feature selection (FS) is a significant topic in data mining and ML. The purpose of feature selection is to select a subset of relevant features to build effective prediction models. Various FS methods have been proposed. Based on the selection criterion choice, these methods can be of three categories: filter methods, wrapper methods, and embedded methods approaches. Filter methods depends on the features of data such as correlation, distance and information, without involving any learning algorithm. Wrapper methods require a predetermined learning algorithm to evaluate the performance of selected features. Embedded methods aim to integrate the feature selection process into the model training process; they are faster than the wrapper methods; and still provide suitable feature subset for the learning algorithm. Online feature selection (OFS) for mining big data has been studied to solve the feature selection problem by an online learning method. The objective of OFS was to develop online classifiers that involve only a small and fixed number of features. Results demonstrate the proposed algorithms are fairly effective for feature selection tasks of online applications, and more efficient and scalable than some state-of-the-art batch feature selection method (HOI et al., 2012).

6. Conclusion

Splitting criteria of decision trees are chosen based on some quality measures, which requires handling the entire data set of each expanding nodes. This makes it difficult for decision trees to be used in big data applications. SVM shows very good performance to data sets in a moderate size. It has inherent limitations to big data applications. Deep learning is suited to address issues related to the volume and variety of big data. However, it has some restrictions in big data because it requires much training time. PLANET can deal with the volume of data, but is not applicable to data with categorical attributes.

Machine learning applications in big data has met challenges such as memory constraint, no support (in iterations) from MapReduce, difficulty in dealing with big data due to Vs (such as high velocity, volume, and variety, etc.), and learning training limited to a certain number of class types or a particular labeled datasets, etc.

Some technology progress has been made such as faceted learning for hierarchical data structure, multi-task learning in in parallel, multi-domain/ cross-domain representation-learning, streaming data processing, high-dimensional data processing, and online feature selection, etc. These areas and the above challenges about machine learning in big data also can be further research topics.

References

Baldominos, A., Albacete, E., Saez Y. & Isasi, P. (2014). A scalable machine learning online service for big data real-time analysis. *2014 IEEE Symposium on Computational Intelligence in Big Data (CIBD): proceedings,* 2014, IEEE, 1-8. DOI: http://dx.doi.org/10.1109/CIBD.2014.7011537

Bu, Y., Borkar, V., Carey, M. J., Rosen, J., Polyzotis, N., Condie, T., Weimer, M. & Ramakrishnan, R. (2012). Scaling datalog for machine learning on big data, March, arXiv:1203.0160v2 [cs.DB].

Chen, C. L. P. & Zhang, C.-Y. (2014). Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Information Sciences, 275*(10), 314-347.

Dean, J. (2014). *Big Data, Data Mining, and Machine Learning: Value Creation for Business Leaders and Practitioners*. John Wiley & Sons, Inc.

Demchenko, Y., Grosso, P., Laat, D. C. & Membrey, P. (2013). Addressing big data issues in scientific data infrastructure. *2013 International Conference on Collaboration Technologies and Systems (CTS),* 20-24 May 2013, San Diego, CA, USA, 48-55.

Hido, S., Tokui, S. & Oda, S. (2013). Jubatus: An open source platform for distributed online machine learning, technical report of the joint jubatus project by preferred infrastructure inc., and ntt software innovation center, Tokyo, Japan, *NIPS 2013 Workshop on Big Learning,* Lake Tahoe. December 9, 2013, 1-6.

HOI, S., Wang, J., Zhao, P. & Jin, R. (2012). Online feature selection for mining big data. *BigMine '12 Proceedings of the 1st International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications,* 2012, ACM New York, NY, USA, 93-100.

Jagadish, H. V., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J. M., Ramakrishnan, R., & Shahabi, C. (2014). Big data and its technical challenges. *Communications of the ACM*, *57*(7), 86-94.

Jaswant, U. & Kumar, P. N. (2015). Big data analytics: a supervised approach for sentiment classification using mahout: an illustration. *International Journal of Applied Engineering Research*, *10*(5), 13447-13457.

Kanagavalli, S., Vaishali, S. & Jeba, J. L. (2015). Analysis and mining of social network data for society issues by using big data. *International Journal of Applied Engineering Research*, 10(4), 10497-10506.

Karacapilidis, N., Tzagarakis M. & Christodoulou, S. (2013). On a meaningful exploitation of machine and human reasoning to tackle data-intensive decision making. *Intelligent Decision Technologies*, *7*, 225–236.

Kotsiantis, S. B. (2007). Supervised machine learning: a review of classification techniques, *Informatica*, *31*, 249-268.

Lee, K. M. (2014). Grid-based single pass classification for mixed big data, *International Journal of Applied Engineering Research*, *9*(21), 8737-8746.

Li, L. (2015). Experimental comparisons of multi-class classifiers. *Informatica, 39*, 71–85.

Low, Y., Gonzalez, J., Kyrola, A., Bickson, D., Guestrin, C. & Hellerstein, J. M. (2012). Distributed graphlab: a framework for machine learning and data mining in the cloud. *The 38th International Conference on Very Large Data Bases*, August 27- 31, 2012, Istanbul, Turkey. *Proceedings of the VLDB Endowment*, 5 (8), 716-727.

Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald R. & Muharemagic, E. (2015). Deep learning applications and challenges in big data analytics. *Journal of Big Data*, *2*(1), DOI 10.1186/s40537-014-0007-7.

Nasridinov, A. (2014). Combining unsupervised and supervised machine learning to analyze crime data. *International Journal of Applied Engineering Research*, *9*(23), 18663-18669.

O'Leary, D.E. (2013). 'Big data', the 'internet of things' and the 'internet of signs'. *Intelligent Systems in Accounting, Finance and Management*, *20*, 53-65.

Qian, H. (2014). PivotalR: a package for machine learning on big data. *The R Journal*, *6*(1), 57-67.

Sukumar, S. R. (2014). Machine learning in the big data era: are we there yet? Conference: *ACM Knowledge Discovery and Data Mining: Workshop on Data Science for Social Good*, Oak Ridge National Laboratory, December 8, 2014, 1-5.

Suthaharan, S. (2014). Big data classification: problems and challenges in network intrusion prediction with machine learning. *Performance Evaluation Review*, *41*(4), 70-73.

Tarwani, K. M., Saudagar, S. S. & Misalkar, H. D. (2015). Machine learning in big data analytics: an overview. *International Journal of Advanced Research in Computer Science and Software Engineering*, *5*(4), 270-274.

Turk, M. (2012). A chart of the big data ecosystem, take 2. http://mattturck.com/2012/10/15/a-chart-of-the-big-data-ecosystem-take-2/

Vu, A. T., De Francisci Morales, G., Gama, J., & Bifet, A. (2014, October). Distributed adaptive model rules for mining big data streams. In *Big Data (Big Data), 2014 IEEE International Conference on* (pp. 345-353). IEEE.

Wang, J.-W., Tang, Y., Nguyen, M. & Altintas, I. (2014). A scalable data science workflow approach for big data bayesian network learning, *BDC '14 Proceedings of the 2014 IEEE/ACM International Symposium on Big Data Computing*. IEEE Computer Society Washington, DC, USA, 16-25.

Zafarani, R., Abbasi, M. A. & Liu. H. (2014). *Social media mining*: *an introduction,* April 20, Cambridge University Press, UK.

Zaslavsky, A., Perera C. & Georgakopoulos, D. (2012). Sensing as a service and big data. *International Conference on Advances in Cloud Computing (ACC),* Bangalore, India, July 2012, 1-8.