Chapter 1
Big Data Analytics and Machine Learning for Industry 4.0: An Overview

Nguyen Tuan Thanh LE [1]
Thuyloi University, 175 Tay Son, Dong Da district, Hanoi, Vietnam, e-mail: thanhlnt@tlu.edu.vn

Manh Linh PHAM [2]
VNU University of Engineering and Technology, Building E3, 144 Xuan Thuy, Cau Giay district, Hanoi, Vietnam, e-mail: linhmp@vnu.edu.vn

Abstract: Our industry is now upgrading to the next industrial revolution, or Industry 4.0, which have been generating massive data that we have never seen before. It requires us to employ new methods to take advantage of this fast and big data. Optimizing and fully automated production, by harnessing cutting-edge technologies, are the ultimate goals of Industry 4.0. Among various advanced and cutting-edge technologies, machine learning (ML) and big data analytics (BDA) have been incorporated and applied successfully to obtain insights from the data and help to adjust automatically industrial processes as needed.

Keywords: Big Data Analytics, Industry 4.0, Machine Learning, Deep Learning

[1] ORCID: 0000-0002-3527-4066, corresponding author
[2] ORCID: 0000-0001-9170-756X, corresponding author

1 Big data analytics for Industry 4.0

1.1 Characteristics of Big data

The concept of "Big data" was mentioned for the first time by Roger Mougalas in 2005 [18]. It refers to a large scale data, one of the characteristics of Industry 4.0, that cannot be stored in a single computer and is almost impossible to be handled using traditional data analytics approaches. Big data applications exploded after 2011 are related to the improvement in computing power, storage as well as the reduction in the cost of sensors, communication and recently the development of Internet of Things (i.e., IoT). These advances have leaded to the utilization of multiple sources (sensors, applications, people, and animals) in the generation of data. In 2011, Big data is defined by [6] using 4Vs characteristics, including: *Volume*, *Velocity*, *Variety*, and also *Value*. Then the fifth one, *Veracity*, was introduced in 2012 [10], as shown in Fig. 1.

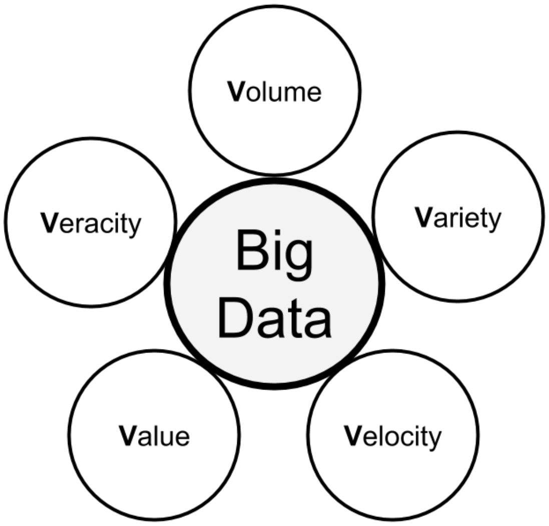

Fig. 1 5Vs characteristics of Big data

Volume hints to the size and/or scale of datasets. Until now, there is not an universal threshold for data volume to be considered as big data, because of the time and diversity of datasets. Generally, big data can have the volume starting from exabyte (EB) or zettabyte (ZB) [4].

Variety implies the diversity of data in different forms which contains *structured*, *semi-structured*, or *unstructured* ones. Real-world datasets, coming from heterogeneous sources, are mostly under unstructured or semi-structured form that make the analysis to be challenged because of the inconsistency, incompleteness, and noise. Therefore, data prepossessing is needed to remove noise, which includes some steps as *data cleaning*, *data integrating*, and *data transforming* [9].

Velocity indicates the speed of processing data. It can fall into 3 categorises: streaming processing, real-time processing, or batch processing. This characteristic emphasizes that the speed of producing data should keep up with the speed of processing data [4].

Value alludes the usefulness of data for decision making. Giant companies (e.g., Amazon, Google, Facebook, etc.) analyze daily large scale datasets of users and their behaviors to give recommendations, improve location services, or provide targeted advertising, etc. [10].

Veracity denotes the quality and trustworthiness of datasets. Due to variety characteristic of data, the accuracy and trust become harder to accomplish and they play an essential role in applications of big data analytics (BDA). As analysing millions of health care entries in order to respond an outbreak that impacts a huge number of people (e.g., CoVid-19 pandemic) or veterinary records to guess the plague in swine herd (e.g., African swine fever or Porcine reproductive and respiratory syndrome), any ambiguities or inconsistencies in datasets can impede the precision of analytic process [10], leading to a catastrophic situation.

Generally, big data in the context of Industry 4.0 can originate from several and various sources, such as: product or machine design data, machine-operation data from control systems, manual-operation records performed by staff, product-quality and process-quality data, manufacturing execution systems, system-monitoring and fault-detection deployments, information on operational costs and manufacturing, logistics information from partners, information from customers on product utilization, feedback, and so on and so forth [25]. Some of these datasets are semi-structured (e.g., manual-operation records), few are structured (e.g., sensor signals), and others are unstructured completely (e.g., images). Therefore, an enterprise 4.0 requires cutting-edge technologies that can fully take advantage of the valuable manufacturing data, including: machine learning (ML) and BDA.

1.2 Characteristics of Big data analytics

BDA can be referred to as "*the process of analyzing large scale datasets in order to find unknown correlations, hidden patterns, and other valuable information which is not able to be analysed using conventional data analytics*" [7]. As the conventional data analysis techniques are no longer effective because of the special characteristics of big data: massive, heterogeneous, high dimensional, complex, erroneous, unstructured, noisy, and incomplete [27].

BDA has attracted attention from not only academic but also industrial scientists as the requirement of discovering hidden trends in large scale datasets increases. [11] compared the impact of BDA for Industry 4.0 with the invention of the microscope and telescope for biology and astronomy, respectively. Recently, the considerable development in the ubiquitous IoT (i.e., Internet of Things), sensor networks, and CPS (i.e., cyber-physical systems) have expanded the data-collection process to an enormous scale in numerous domains, including: social media, smart cities, education, health care, finance, agriculture, etc. [10].

Various advanced techniques to analyze data (i.e., ML, computational intelligence, data mining, natural language processing) and potential strategies (i.e., parallelisation, divide and conquer, granular computing, incremental learning, instance selection, feature selection, and sampling) can help to handle big data issues. Empowering more efficient processing, and making better decisions can also be obtained by using these techniques and strategies [10].

Divide and conquer helps to reduce the complexity of computing problem. It is composed of three phases: firstly, it reduces the large-complex problem into several smaller-easier ones; secondly, it tries to solve each smaller problem; and finally, it combines solutions of all smaller problems to solve the original problem [10].

Parallelisation allows to improve computation time by dividing big problems into smaller instances, distributing smaller tasks across multiple threads and then performing them simultaneously. This strategy decreases computation time instead of total amount of work because multiple tasks can be performed simultaneously rather than sequentially [28].

Incremental learning is widely practiced and used to handle streaming data. It is a learning algorithm and can be trained continuously with additional data rather than current ones. In learning process, this strategy tunes parameters each time new input data comes [28].

Granular computing helps to simplify the elements from a large space by grouping them into subsets, or granules [2, 12]. By reducing large elements to a search space which is smaller, uncertainty of elements in this search space is identified effectively [30].

Feature selection is useful for preparing high scale datasets. This strategy handles big data by determining a subset of relevant features which are for an aggregation. Nevertheless, the data representation is more precise in this particular strategy [16].

Instance selection is a major approach for pre-processing data. It helps to shorten training sets and run-time in the training phases [21].

Sampling is a method for data reducing that helps to derive patterns in big datasets by generating, manipulating, and analyzing subsets of the original data [28].

2 Machine learning for Industry 4.0

Machine learning (ML), a state-of-the-art subfield of Artificial Intelligence (AI) – Fig. 4, has been now powering several aspects of our society: information search on the Internet, content filtered on social networks, recommendations of e-commercial platforms, or accurate language translation, virtual classrooms in education, support for diagnosing diseases in medicine, etc. [13]. It has been applied successfully to solve several real problems, such as: transcribing speech into text, matching new items, identifying objects, selecting relevant search results, etc. [13].

Actually, the goal of a typical ML to find a *mathematical formula* (i.e., the model), when applied to a collection of inputs (i.e., the training data) then produces the desired outputs [3]. The “invented" mathematical formula is also expected to generate the “correct" outputs for most other new inputs (distinct from the training data) on the assumption that those inputs come from the *same* or a *similar* statistical distribution of the training data [3].

In order to teach the machine, three components are needed, including: (1) *data* - the more diverse and bigger the data, the better the result; (2) *features* - also know as *parameters* or *variables* (e.g., age, gender, stock price, etc.), they are the factors that the machine is looking at; and (3) *algorithms* - the steps we follow to solve the given problem that affects the precision, performance, and size of our model [32].

Generally, ML algorithms can be classified into 4 main types: (1) *Unsupervised learning*, (2) *Semi-supervised learning*, (3) *Supervised learning*, and (4) *Reinforcement learning* [5, 3], as shown in Fig. 2.

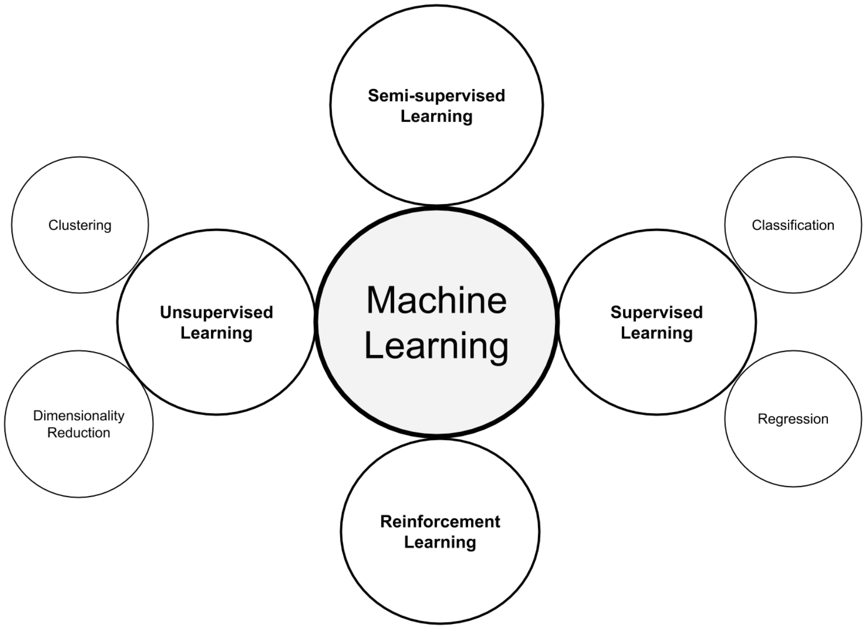

Fig. 2 Four main categories of Machine Learning

2.1 Supervised learning

Find out a mathematical formula that maps inputs to *already-known* outputs, provided a set of human-annotated examples, is the purpose of a supervised learning algorithm. In this case, we have a “supervisor" or a “teacher" who gives the machine all the answers, like whether it’s a cat or a dog in a given picture [32], i.e., a *classification* problem. The “teacher" has already labeled input datasets and the machine will learn on top of these examples [32].

A specific kind of supervised learning is *self-supervised learning*, in which machine learns without human-annotated labels [5]. There are still associated labels, but they are generated from the input data typically using a heuristic algorithm [5].

2.2 Unsupervised learning

In contrast to the former, in this case, the machine has no “supervisor" or “teacher". Input data is not labeled, the machine is left on its own, trying to find certain hidden patterns or structures in datasets. For example, in *clustering* problem, the model (i.e., a mathematical formula) will output a cluster identifier for each input data. Or in *dimensionality reduction* problem, the model will output a new vector with fewer features than the original one of input data [3].

2.3 Semi-supervised learning

In this case, input data contains not only labeled but also unlabeled examples. The purpose of a semi-supervised learning is the same as the one of the supervised learning [3]. By using several unlabeled examples (i.e., adding more information about the problem and then reflects better the probability distribution of data), we can help our learning algorithm to find out better models [3].

2.4 Reinforcement learning

In this case, the machine, also called an *agent*, is embedded in an environment. It is capable of observing *states* of that environment as a vector of features, and then can perform *actions* in response to these states [3]. Different actions can give different *rewards* to the agent and could also move it to another state of the environment, and so on. In contrast to supervised learning and unsupervised learning, where we operate with *static* datasets, in reinforcement learning, we work with *dynamic* datasets collected repeatedly from a dynamic environment.

Learning an *optimal policy* is the target of an reinforcement learning algorithm, i.e., a function that inputs the feature vector of a state and then outputs an optimal sequence of actions to execute in that state [3], in order to maximize long-term accumulated reward. For instance, sequence of scaling actions such as adding or removing virtual machines/containers to keep up with fluctuation of resource's demand for big-data-analytic application can be a result drawn from a reinforcement learning algorithm. Feature vector here consists of information from the application itself (e.g., current computing power, workload) and surrounding environment (e.g., type of media, other co-located applications).

2.5 Machine learning for Big data

Conventional ML approaches cannot handle efficiently big data problems because of its 5V's characteristics (i.e., high speeds, diverse sources, low value density, large volumes and incompleteness) [10]. Therefore, several advanced ML techniques for BDA are proposed: *transfer learning*, *feature learning*, *active learning*, *distributed learning*, and *deep learning* [10].

Feature learning empowers a system to figure out automatically the representations required for feature detection or classification from raw datasets [10].

Transfer learning allows to employ knowledge which has been learned from one context to new contexts. By transferring useful information from similar domains, it efficiently improves a learner from one specific domain [29].

Distributed learning aims to alleviate the scalability issue of conventional ML by distributing computations on datasets among a couple of machines for scaling up the process of learning [23]. One of platforms using this distributed approach to resolve scaling problem for multi-cloud applications was proposed by [22].

Active learning aims to employ adaptive data collection. In this process, parameters are adjusted automatically to gather as quickly as possible useful data for accelerating ML tasks and overpowering the problem of labeling [1].

Deep learning (DL) can be employed to extracts complex and high-level abstractions of data representations. It is done by using a hierarchical, layered architecture of learning, where more abstract features (i.e., higher-level) are stated, described, and implemented on top of less abstract ones (i.e., lower-level) [20] – see Fig. 3(a). DL techniques can analyze and learn from enormous amount of unsupervised data, which is suitable for BDA in which raw data is mostly unlabeled as well as uncategorised [20]. We will focus on DL for Industry 4.0 on the next section.

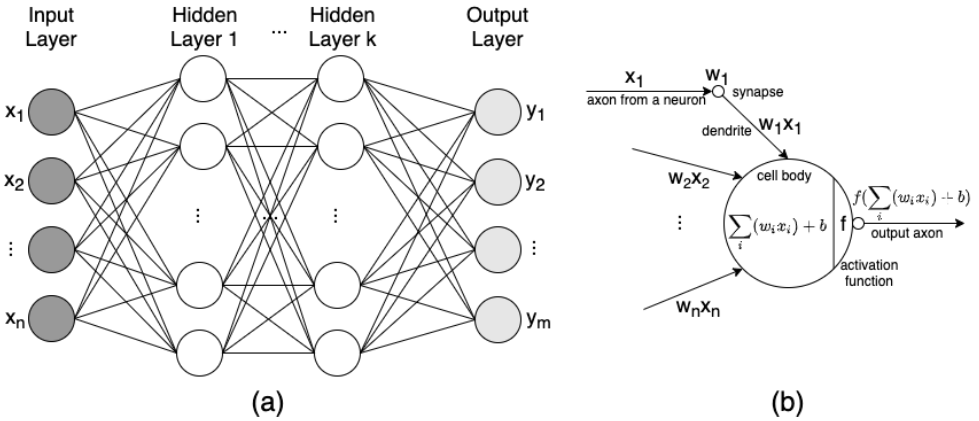

Fig. 3 (a) Typical architecture of deep learning neural network with one output, one input, and k hidden layers; (b) Artificial neuron: basic computational building block for neural networks

3 Deep learning for Industry 4.0: State of the art

Deep learning, the most exciting branch of ML – Fig. 4, has been expanded on the base of classic Artificial Neural Networks (NNs). It supports computational models that, in contrast to *shallow* NN-like models with only few layers, are consisted of multiple processing (non-linear) layers. Each layer will take charge of different level of abstraction that helps to learn hierarchical representation of data. The functionality of DL is emulated from the operation of neuron network in human brain for processing ambient signals [19], with the notions of *axon*, *synapse*, *dendrite* – see Fig. 3(b). DL, with different types (e.g., Recurrent Neural Networks, Autoencoders, Convolutional Neural Networks, Deep Belief Net, etc.), has outperformed others conventional ML techniques as well as improved dramatically cutting-edge real problems in recognition of object, speech recognition, object detection, language translation and several other areas such as self-driving car, genomics, games, or drug discovery, etc. [24, 13].

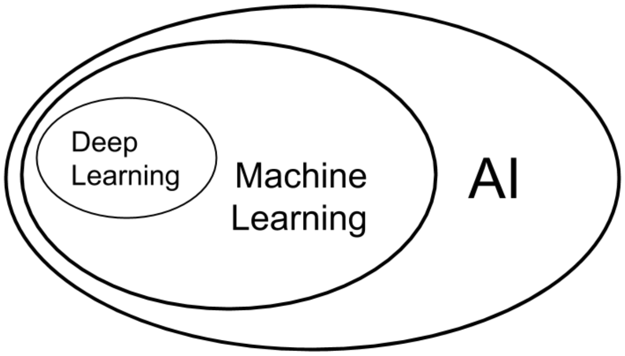

Fig. 4 Relationships between DL, ML, and AI [8]

DL helps to discover convoluted structure in a large scale dataset by using an optimization algorithm, called *backpropagation*, meaning “error backward propagation". It specifies how a model changes its (up to billions of) internal parameters. In each layer, these parameters are used to compute the representation based on the last one [13]. Most contemporary DL algorithms are based on Stochastic Gradient Descent (SGD) [8].

In addition, DL requires fairly manual engineering, as it can profit from increases in the amount of available data and computation [13], thus suitable for BDA. In consequence, several hidden features, that might not be seen obviously by a human, can be exposed by using a DL model [19].

For the industry sector, in order to accelerate technologies toward smart manufacturing, equipping intelligent and high-precision systems, is very important because they affect straightly to efficiency of related products, reinforced productivity, and also reduce operation costs as well as maintenance expenses [19]. In this context, a DL model can play an essential role. Indeed, a wide form of applications for industry as controlling robots, object detection and tracking, visual inspection of product lines, fault diagnosis, etc., can benefit by applying a DL model [19].

Luckow et al. [17] investigated visual inspection of product lines using Convolutional Neural Network architectures including AlexNet and GoogLeNet over different DL platforms, such as: Tensorflow, Caffe, and Torch. In this work, several vehicles images, along with their annotation, in the assembly line are submitted to a DL system. Consequently, by using Tensorflow platform, they achieved the best performance with accuracy of 94%.

Lee et al. [15] tackled with detection of faults found out in the process of transferring geometric shapes on a mask to the surface of a silicon wafer and classification problem in noisy settings by employing Stacked Denoising Auto-Encoders (SdA). It helps to lower the noise contained in descriptive sensory data, derived from electrically mechanic disturbances as well as carry out classification of fault. Results of this paper showed that, in comparison with other baseline methods (e.g., Support Vector Machine or K-Nearest Neighbors), SdA drives to about 14% more accuracy in noisy situations.

Another work involved SdA was of Yan et al. [31]. They performed the detection of abnormal actions of a combustion gas turbines by applying extreme learning machines joint with SdA. Their results showed that the features detected by SdA leaded to a more improved classification in comparison with hand-crafted features.

Shao et al. [26] extracted features in a fault diagnosis system for rotating devices with the input of vibration data by applying Deep Neural

Networks. The authors combined Denoising Auto-Encoders with Contractive Auto-Encoders. To diagnose the fault, they refined the learned features using Locality Preserving Projection, then put them into a softmax classifier. Seven conditions were considered in their system, including: *rubbing fault*, *compound faults* (rub and unbalance), *4 levels of imbalance faults* as well as *normal operation*. The device status is identified based on exploitation of vibration data by the diagnosis system. It figures out whether the device is in fault or normal condition. Their approach used on the experiments to diagnose the fault of locomotive bearing devices and rotors was shown that it can beat Convolutional Neural Network and other shallow learning methods.

Lee [14] supported detection of faults belong to several defect types often appear on headlight modules of cars in a setting of vehicle manufacturer by proposing a Deep Belief Network (DBN) model together with a cloud platform and an IoT deployment. The results showed that DBN model outperformed two other baseline methods (i.e., Radial Basis Function, and Support Vector Machine) with regard to error rate in test datasets.

4 Conclusion

In this chapter, we have reviewed two promising technologies for Industry 4.0, named *BDA* and *ML*. We focus on the data aspect of smart manufacturing, which is fast and massive, and cannot be handled efficiently by conventional approaches. Indeed, by employing BDA and ML, especially DL, a wide range of industrial applications is proven to be accelerated. Although few successful works were reported in the literature, we believe that an optimizing and fully automated production on a large scale could be achieved in a very-near future because of these potential advanced technologies.

Acknowledgements

This work has been partly supported by Vietnam National University, Hanoi (VNU), under Project No. QG.20.55.

References

[1] S. Athmaja, M. Hanumanthappa, and V. Kavitha. A survey of machine learning algorithms for big data analytics. In *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pages 1–4. IEEE, 2017.

[2] A. Bargiela and W. Pedrycz. Granular computing. In *Handbook on Computational Intelligence: Volume 1: Fuzzy Logic, Systems, Artificial Neural Networks, and Learning Systems*, pages 43–66. World Scientific, 2016.

[3] A. Burkov. *The hundred-page machine learning book*. Andriy Burkov Quebec City, Can., 2019.

[4] M. Chen, S. Mao, and Y. Liu. Big data: A survey. *Mobile networks and applications*, 19(2):171–209, 2014.

[5] F. Chollet. *Deep Learning with Python*. Manning Publications Co., 2017.

[6] J. Gantz and D. Reinsel. Extracting value from chaos. *IDC iview*, 1142(2011):1–12, 2011.

[7] N. Golchha. Big data-the information revolution. *Int. J. Adv. Res*, 1(12):791–794, 2015.

[8] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

[9] J. Han, J. Pei, and M. Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.

[10] R. H. Hariri, E. M. Fredericks, and K. M. Bowers. Uncertainty in big data analytics: survey, opportunities, and challenges. *Journal of Big Data*, 6(1):44, 2019.

[11] M. Hilbert. Big data for development: A review of promises and challenges. *Development Policy Review*, 34(1):135–174, 2016.

[12] J. Kacprzyk, D. Filev, and G. Beliakov. *Granular, soft and fuzzy approaches for intelligent systems*. Springer, 2017.

[13] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[14] H. Lee. Framework and development of fault detection classification using iot device and cloud environment. *Journal of Manufacturing Systems*, 43:257–270, 2017.

[15] H. Lee, Y. Kim, and C. O. Kim. A deep learning model for robust wafer fault monitoring with sensor measurement noise. *IEEE Transactions on Semiconductor Manufacturing*, 30(1):23–31, 2016.

[16] H. Liu and H. Motoda. *Computational methods of feature selection*. CRC Press, 2007.

[17] A. Luckow, M. Cook, N. Ashcraft, E. Weill, E. Djerekarov, and B. Vorster. Deep learning in the automotive industry: Applications and tools. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 3759–3768. IEEE, 2016.

[18] R. Magoulas and B. Lorica. Introduction to big data. *Radar. Release*, 2, 2009.

[19] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani. Deep learning for iot big data and streaming analytics: A survey. *IEEE Communications Surveys & Tutorials*, 20(4):2923–2960, 2018.

[20] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic. Deep learning applications and challenges in big data analytics. *Journal of Big Data*, 2(1):1, 2015.

[21] J. A. Olvera-López, J. A. Carrasco-Ochoa, J. F. Martínez-Trinidad, and J. Kittler. A review of instance selection methods. *Artificial Intelligence Review*, 34(2):133–143, 2010.

[22] L. M. Pham and T.-M. Pham. Autonomic fine-grained migration and replication of component-based applications across multi-clouds. In *2015 2nd National Foundation for Science and Technology Development Conference on Information and Computer Science (NICS)*, pages 5–10. IEEE, 2015.

[23] J. Qiu, Q. Wu, G. Ding, Y. Xu, and S. Feng. A survey of machine learning for big data processing. *EURASIP Journal on Advances in Signal Processing*, 2016(1):67, 2016.

[24] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[25] M. Schuldenfrei. Big data challenges of industry 4.0, 2019.

[26] H. Shao, H. Jiang, F. Wang, and H. Zhao. An enhancement deep feature fusion method for rotating machinery fault diagnosis. *Knowledge-Based Systems*, 119:200–220, 2017.

[27] C.-W. Tsai, C.-F. Lai, H.-C. Chao, and A. V. Vasilakos. Big data analytics: a survey. *Journal of Big data*, 2(1):21, 2015.

[28] X. Wang and Y. He. Learning from uncertainty for big data: future analytical challenges and strategies. *IEEE Systems, Man, and Cybernetics Magazine*, 2(2):26–31, 2016.

[29] K. Weiss, T. M. Khoshgoftaar, and D. Wang. A survey of transfer learning. *Journal of Big data*, 3(1):9, 2016.

[30] R. R. Yager. Decision making under measure-based granular uncertainty. *Granular Computing*, 3(4):345–353, 2018.

[31] W. Yan and L. Yu. On accurate and reliable anomaly detection for gas turbine combustors: A deep learning approach. *arXiv preprint arXiv:1908.09238*, 2019.

[32] V. Zubarev. Machine learning for everyone: In simple words. with real-world examples. yes, again, 2019.