Designing more accurate machine-learning systems so as to satisfy the market needs will hence lead to a higher likelihood of energy waste due to the increased computational cost.

Nowadays, there is a greater need to develop efficient intelligent models to cope with future demands that are in line with similar energy-related initiatives. Such energy-efficient-oriented data modeling is important for a number of data-intensive areas, as they affect many related industries. Designers should focus on maximum performance and minimum energy use so as to break away from the traditional' performance vs. energy-use' tradeoff, and increase the number and diversity of options available for energy-efficient modeling. However, despite the fact that there is a demand for such efficient and sustainable data modeling methods for large and complex data-intensive fields, to our best knowledge, only a few of these literatures have been proposed in the field [6,7].

This paper provides a comprehensive review of state-of-the-art sustainable/energy-efficient machine-learning literatures, including theoretical, empirical and experimental studies pertaining to the various needs and recommendations. Our objective is to introduce a new perspective for engineers, scientists, and researchers in the computer science, and green ICT domain, as well as to provide its roadmap for future research endeavors.

This paper is organized as follows. Section 2 introduces the different large-scale data-intensive areas and discusses their structure and nature, including the relation between data models and their characteristics. Section 3 discusses the issues in current intelligent data modeling for sustainability and gives recommendations. Section 4 concludes the paper.

2. Big data challenge

e-Science areas are typically data-intensive in that the quality of their results improves with both quantity and quality of data available. However, current intelligent machine-learning systems are not inherently efficient enough which ends up, in many cases, a growing fraction of this quantity data unexplored and underexploited. It is no small problem when existing methods fail to capture such data immensity. When old concepts fail to keep up with change, traditions and past experience become inadequate guide for what to do next. Effective understanding and the use of this new wealth of raw information pose a great challenge to today's green engineers/researchers. It should be noted that the scope of the review is limited to the analytical aspects of science areas using immense datasets, and the methods for reducing computational complexity in distributed or grid-computing environment are excluded.

2.1. Geo, climate and environment

There are many recent examples that can illustrate the tremendous growth in scientific data generation in the literature. It is estimated that there are thousands of wireless sensors currently in place, which generates about a gigabyte of data per sensor per day [8]. Such sensors measure and record sensory information about the natural environment at a joint spatial and temporal dimensions that has never previously been possible. This environmental information is gathered by sensors via its sensing devices that are attached to small, low-power computer systems with digital radio communications. The sensor nodes self-organize itself into a network to deliver, and perhaps process the collected data to a base station, where it can be made available to the users through the Internet. These sensors generate several petabytes of data per year and decisions need to be taken in real time as to how much data to analyze, how much to transmit for further analysis.

Besides the environmentalists, a similar challenge facing the climatologists, meteorologists, and geologists today is also making sense of the vast and continually increasing amount of data generated by the earth observation satellites, radars, and high-throughput sensor networks. The World Data Centre for Climate (WDCC) is the world-largest climate data repository, and is also known to have the largest database in the world [9]. The WDCC archives 340 terabytes of earth system model data and related observations, and 220 terabytes of data readily accessible on the web including information on climate research and anticipated climatic trends, as well as 110 terabytes (or 24,500 DVD's) worth of climate simulation data. The WDCC data is accessible by a standard web-interface (http://cera.wdc-climate.de). These data are increasingly available in many different formats and have to be incorporated correctly into the various climate change models. Timely and accurate interpretation of these data can provide advance warnings in times of severe weather changes, hence enabling corresponding action to be taken promptly so as to minimize its resulting catastrophic damage.

2.2. Bio, medicine, and health

Biological data has been produced at a phenomenal rate due to the international research effort called the Human Genome Project. It is estimated that the human genome DNA contains around 3.2 billion base (3.2 gigabase) pairs distributed among twenty-three chromosomes, which is translated to about a gigabyte of information [10]. However, when we add the gene sequence data (data on the 100,000 or so translated proteins and the 32,000,000 amino acids), the relevant data volume can easily expand to an order of about 200 gigabyte [11]. Now, by including also the X-ray/NMR spectroscopy structure determination of these proteins, the data volume will increase dramatically to several petabytes, and that is assuming only one structure per protein.

As of December 2014, the GenBank repository of nucleic acid sequences contained above 178 million entries [12] and the SWISS-PROT database (inc. both UniProtKB/Swiss-Prot, UniProtKB/TrEMBL) of protein sequences contained about 18 million entries [13,14]. On average, these databases are doubling in size in every 15 months. This is further compounded by data generated from the myriad of related projects that study gene expression, that determines the protein structures encoded by the genes, and that details how these proteins interact with one another. From that, we can begin to imagine the enormous amount and variety of information that is being produced every month.

Over the past decade, the health sector has also evolved significantly, from paper-based systems to largely paperless electronic systems. Many countries' public health systems are now providing electronic patient records with advanced medical imaging media. In fact, this has already been implemented by more than 200 American hospitals, and the days of squinting to decipher a doctor's untidy scrawl on a handwritten prescription will soon be a thing of the past in Canada and many other countries too [15].

InSiteOne is one of the leading service providers in offering data archiving, storage, and disaster-recovery solutions to the healthcare industry. Its U.S. InSiteOne's archives include almost 4 billion medical images and 60 million clinical studies, in a coverage area of about 800 clinical sites [16]. The combined annual total of its radiological images exceeds 420 million and this number is still increasing at an approximate rate of about 12% per year. There are about 35,500 radiologists currently practicing in the U.S. [17]. Each image will typically constitute several megabytes of digital data and is required to be archived for a minimum of five years. ESG (Enterprise Storage Group) forecasts medical image data in North America will grow to more than 35 percent per year and will reach nearly 2.6 million terabytes by 2014 [18]. It is also worthwhile to

Please cite this article in press as: O.Y. Al-Jarrah et al., Efficient Machine Learning for Big Data: A Review, Big Data Research (2015), http://dx.doi.org/10.1016/j.bdr.2015.04.001

note that for the digital health data, its integrity and security issues are of critical importance in the field. For instance, for the former, data compression techniques may not be used, in many cases, as they may distort the data; and for the latter, the confidentiality of patient data is clearly cardinal in order to foster public confidence in such technologies.

2.3. Stars, galaxies, and the universe

The digital data volume from the stars, galaxies and universe has multiplied over the past decade due to the rapid development of new technologies such as new satellites, telescopes and other observatory instruments. Recently, the Visible and Infrared Survey Telescope for Astronomy (VISTA) [19] and the Dark Energy Survey (DES) [20] – the largest universe survey projects initiated by two different consortiums of universities, from the UK, and from the U.S., are expected to yield databases of 20–30 terabytes in size in the next decade.

According to DES, its observatory field is so large that a single image will record data from an area of the sky 20 times the size of the moon as seen from the earth [20]. The survey will image 5000 degrees of the U.S. southern sky and will take about five years to complete. As for VISTA, its performance requirements were so challenging that it peaks at 55 megabytes/second data rate with a maximum of 1.4 terabytes of data per night [19]. But, these are now fairly commonplace. Many other astro-scientific databases, such as the Sloan Digital Sky Survey (SDSS) are already terabytes in size [21] and the Panoramic Survey Telescope-Rapid Response System (Pan-STARRS) is expected to produce a science database of more than 100 terabytes in size for the next five years [22]. Likewise, the Large Synoptic Survey Telescope (LSST) is producing 30 terabytes of data per night, yielding a total database of about 150 petabytes [23]. As the data produced by the new telescopes are expected to come to the Internet, this picture will change radically.

Many believe that the massive data volume and the ever increasing computing power will dramatically change the way in how conventional science and technology are conducted. We believe that this surge in data will open up and challenge further research in each field, hence, instigating the search for new approaches. Likewise, such challenge needs to be addressed in the area of intelligent information science as well.

3. Sustainable data modeling and efficient learning

With consideration of the large influx of data, it is definitely necessary to improve the way in how conventional computational/analytic data models are designed and developed. Sustainable data modeling can be defined as a form of data modeling technology, aimed to make sense of the large amount of data associated in its own field, by discovering patterns and correlations in an effective and efficient way. Sustainable data modeling specifically focuses on 1) maximum learning accuracy with minimum computational cost, and 2) rapid and efficient processing of large volumes of data. Sustainable data modeling seems to be ideal because of its ease in which large quantities of data are handled efficiently as well as its associated cost reduction observed in many cases. In a wider perspective, it entails a data-modeling revolution in e-sciences. In fact, these newly designed sustainable data models will effectively cope with the above data issues and, as a result, bring about benefits to the various e-science areas. Some of the excellent examples are well discussed in Patnaik et al., Sundaravaradan et al., and Marwah's article [24–27]. Hence, in this section, we will give a few recommendations to green engineers/researchers on a few key mechanics of the sustainable data modeling.

3.1. Ensemble models

One of the key success elements of sustainable data modeling is to maintain or improve its performance while significantly reducing its computational cost. Recent data-modeling research has shown that ensemble methods have gained much popularity as they often perform better than individual models [28,29]. Ensemble method uses multiple models to obtain better performance than those that could be obtained from any of the constituent models [29,30]. However, it can result in significant increase in computational cost. If the model deals with large-scale data, model complexity and computational requirements will grow exponentially. An example of such ensemble model is the Bayes classifier [31]. In Bayes classifier, each hypothesis is given a vote proportional to the likelihood that the training dataset would be sampled from a system if that hypothesis was true. To facilitate the training data of finite size, the vote of each hypothesis is also multiplied by the prior probability of that hypothesis. The Bayes classifier is expressed as follows:

$$y = \arg\max_{c_j \in C} \sum_{h_i \in H} P(c_j|h_i)P(T|h_i)P(h_i),$$

where y is the predicted class, C is the set of all possible classes, H is the hypothesis space, P refers to a probability, and T is the training data. As an ensemble, the Bayes classifier represents a hypothesis that is not necessarily in H. The hypothesis represented by the Bayes classifier, however, is the optimal hypothesis in ensemble space (the space of all possible ensembles consisting only of hypotheses in H).

Considering the problem of numerical weather prediction, ensemble predictions are now commonly made at most of the major operational weather prediction facilities worldwide [32], including the National Centers for Environmental Prediction, U.S., the European Centre for Medium-Range Weather Forecasts (ECMWF), the United Kingdom Met Office, Metro France, Environment Canada, the Japanese Meteorological Agency, the Bureau of Meteorology, Australia, the China Meteorological Administration, the Korea Meteorological Administration, and CPTEC, Brazil.

3.2. Model complexity problem

Bayes estimation techniques have been well-adopted in general intelligent data modeling because they provide a fundamental formalism for combining all the information available, with regards to the parameters to be estimated, with optimized time complexity [33].

One of the most serious problems in Bayes nonparametric learning models is its high-algorithmic complexity and extensive memory requirements, especially for the necessary quadratic programming in large-scale tasks. As a nonparametric Bayes classifier extracts worst-case example x and uses statistical analysis to build a classifying model, any learning algorithm that examines every attribute values of every training example must have at least the same or worse complexity [33].

Many applications of machine learning deal with problems where both the number of features i as well as the number of examples x_i is large. Linear Support Vector Machines are among the most prominent machine-learning techniques for such high-dimensional and sparse data. In this article, we use two machine-learning models as examples to be semiparameterized. In other words, the two models are to be modified to be more efficient and fast computationally. The time complexity of the Bayes and SVMs are well discussed in Elkan's and Joachims' article respectively [34, 35].

Please cite this article in press as: O.Y. Al-Jarrah et al., Efficient Machine Learning for Big Data: A Review, Big Data Research (2015), http://dx.doi.org/10.1016/j.bdr.2015.04.001

3.3. Local learning strategy

Yoo et al. have proposed two different support-vector-based efficient ensemble models that have shown to reduce its computational cost while maintaining its performance [36]. Their novel learning technique has proven to be successful by other similar studies [7]. With a nonparametric model, a unique model must be constructed for each test set, which will significantly increase its computational complexity and cost.

To reduce the computational cost, they have thus proposed to partition the training samples into clusters, with that, build a separate local model for each cluster – this method is called local learning. A number of recent works have demonstrated that such a local learning strategy is far superior to that of the global learning strategy, especially on data sets that are not evenly distributed [37–40]. If a local-learning method is adopted in the decision function of a nonparametric classifier (i.e., the general regression network), it will allow for the classifier to be semiparameterized. Its semiparametric approximation can be expressed as follows:

$$Z_i \exp \frac{-(x-c_i)^T(x-c_i)}{2\sigma^2} \approx \sum_{j=1}^{Z_i} \exp \frac{-(x-x_j)^T(x-x_j)}{2\sigma^2},$$

where x_i is a training vector for class i in the input space, σ is a single learning or smoothing parameter chosen during the network training, and Z_i is a number of input training vectors x_i associated with its center c_i. In nonparametric classification, many different types of radial basis functions can be chosen in place of the Gaussian function. The radial basis function, used in many cases, is actually a spherical kernel function, which is specifically used for nonparametric function estimation. If the number of training samples approaches infinity, the nonparametric function estimation hence becomes no longer dependent on the parameters of the radial basis function, however, for finite training samples, we can always observe some forms of dependency on the radial basis function parameters.

The local learning strategy provides more dependence on the radial basis function parameters than that of a nonparametric model because the local learning model is a semiparametric approximation of a nonparametric/global learning model. In other words, in semiparametric modeling, model assumptions get stronger than those of nonparametric models, but are less restrictive than those of parametric model. In particular, this approximation avoids the practical disadvantages of nonparametric methods at the expense of increased risk of specification errors. Semiparametric models that are based on local learning help not only in reducing the model complexity but also in finding the optimal tradeoff between the parametric and nonparametric models – so as to achieve both low model bias and variance [41]. In short, it can therefore take on the inherent advantage of both the models while reducing its computational requirements effectively.

3.4. Semiparametric approximation

The above examples can be seen as a spherical function mixture model with data-directed center vector allocation. That is because the relative widths of the spherical functions at each center are directly proportional to the relative number of training vectors associated with each center. Many different types of computational local models, and the diverse selection method of the y_i and the grouping of the associated input vectors in each class i can be used for the global model semiparametric approximation.

The local learning strategy provides a reasonable approximation since x_i are sufficiently close in the input vector space. In that case, they can be adequately represented by a single center vector c_i in that local space. In the case of Support Vector Regression (SVR), the c_i vectors can be derived from either the k-means or the codebook theory. In SVR, where the two classes are not separable, they map the input space into a high-dimensional feature space (where the classes are linearly separable), using a nonlinear kernel function. The kernel function calculates the scalar product of the images of two examples in the feature space.

Given an n-dimensional input vector, $x_i = (x_1, x_2, \ldots, x_n)$ with two labels, $y_i \in \{+1, -1\}$ where $i = 1, 2, \ldots, N$, the hyperplane decision function of the binary SVR with kernel method is:

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{\ell} y_i a_i \langle \Phi(x), \Phi(x_i)\rangle + b\right)$$

$$= \operatorname{sgn}\left(\sum_{i=1}^{\ell} y_i a_i k(x, x_i) + b\right)$$

and the quadratic program is given as:

$$\text{maximize } W(a) = \sum_{i=1}^{\ell} a_i - \frac{1}{2}\sum_{i,j=1}^{\ell} a_i a_j y_i y_j k(x_i, x_j),$$

$$\text{subject to } \quad a_i \geq 0,\ i = 1, \ldots, \ell, \quad \text{and} \quad \sum_{i=1}^{\ell} a_i y_i = 0,$$

where ℓ is the number of training patterns, a_i is the parameters of SVR, $K(.,.)$ is a spherical (nonparametric) kernel function, and b is the bias term. In the above case, the local model can be constructed from k-means clustering. The objective function of the k-means clustering can be expressed as follows:

$$\min_{C,Z} \sum_{j=1}^{k}\sum_{i=1}^{n} Z_{i,j}\|X_i - C_j\|_2^2 + R\sum_{j=1}^{k}\left|\sum_{i=1}^{n} Z_{i,j} y_i\right|,$$

where X_i is the ith row of the similarity matrix $\sum$, C_j is a $1 \times m$ row vector representing the centroid of the jth cluster, R is a nonnegative scaling parameter, and $Z_{ij} \in \{0, 1\}$ is an element of the cluster membership matrix, whose value is equal to one if the ith source vector belongs to the jth cluster, and zero if otherwise. The first term in the objective function corresponds to a cluster cohesion measure. The minimization of the above equation would ensure that the training vectors in the same cluster have highly correlated similarity vectors. The second term measures the skewness of class distribution in each cluster. The minimization of this term would ensure that each cluster contains a balanced number of positive and negative estimation vectors. The cluster centroid C and cluster membership matrix Z are estimated iteratively as follows:

- We fix the cluster centroids and use them to determine the cluster membership matrix.
- The revised cluster membership matrix is used to update the centroids – repeated until the algorithm converges to a local minimum.

To compute the cluster membership matrix Z, we transform the original optimization problem, using k slack variable t_j, into:

Please cite this article in press as: O.Y. Al-Jarrah et al., Efficient Machine Learning for Big Data: A Review, Big Data Research (2015), http://dx.doi.org/10.1016/j.bdr.2015.04.001

$$\min_{Z,t} \sum_{j=1}^{k} \sum_{i=1}^{n} Z_{i,j} \|X_i - C_j\|_2^2 + R \sum_{j=1}^{k} t_j,$$

$$\text{s.t.} \quad -t_j \le \sum_{i=1}^{n} Z_{i,j} y_i \le t_j,$$

$$t_j \ge 0,\ 0 \le Z_{i,j} \le 1,$$

$$\sum_{j=1}^{k} Z_{i,j} = 1,$$

if the cluster membership matrix is obtained, the cluster centroid C_j is updated based on the following:

$$Q_j(X_m) = C_j = \frac{\sum_{i=1}^{n} Z_{i,j} X_i}{\sum_{i=1}^{n} Z_{i,j}}, \quad j = 1, 2, \ldots, N.$$

To construct a semiparametric model, we substituted $Q_i(X)$ for each training sample x_i used in the SVR decision function. The new semiparametric model's approximation is therefore expressed as:

$$f(x) = \operatorname{sgn}\big(w \cdot \phi(x) + b\big) = \operatorname{sgn}\left(\sum_{i=1}^{\ell} y_i a_i k(x, c_i) + b\right),$$

and the quadratic program is given as:

$$\text{maximize } W(a) = \sum_{i=1}^{\ell} a_i - \frac{1}{2} \sum_{i,j=1}^{\ell} a_i a_j y_i y_j k\big(Q_i(x), Q_j(x)\big),$$

$$\text{subject to} \quad a_i \ge 0,\ i = 1, \ldots, \ell, \quad \text{and} \quad \sum_{i=1}^{\ell} a_i y_i = 0.$$

As mentioned, the local model can also be constructed from the principle of codebook [42]. In this case, its basic idea is to replace key values from an original multidimensional vector space with values from a discrete subspace of lower dimension. The lower-dimension vector requires less storage space and the data is thus compressed.

Consider a training sequence consisting of M source vectors, $T = \{x_1, x_2, \ldots, x_m\}$. M is assumed to be sufficiently large, such that all the statistical properties of the source are captured by the training sequence. We assume that the source vectors are k-dimensional, $X_m = (x_{m,1}, x_{m,2}, \ldots, x_{m,k})$, $m = 1, 2, \ldots, M$. These vectors are compressed by choosing the nearest matching vectors, and form a codebook comprising of the entire set of codevectors. N is the number of codevectors, $C = \{c_1, c_2, \ldots, c_n\}$ and each codevector is k-dimensional, $c_n = (c_{n,1}, c_{n,2}, \ldots, c_{n,k})$, $n = 1, 2, \ldots, N$. The representative codevector is determined to be the closest in Euclidean distance from the source vector. The Euclidean distance is defined by:

$$d(x, c_i) = \sqrt{\sum_{j=1}^{k} (x_j - c_{ij})^2},$$

where x_j is the jth component of the source vector, c_{ij} is the jth component of the codevector c_i, S_n is the nearest-neighboring region associated with codevector c_n, and the partitions of the whole region are denoted by $P = \{S_1, S_2, \ldots, S_N\}$. If the source vector X_m is in the region S_n, its approximation can be denoted by $Q(X_m) = c_n$, *if* $X_m \in S_n$. The Voronoi region is defined by:

$$V_i = \big\{x \in R^k\colon \|x - c_i\| \le \|x - c_j\|, \text{ for all } j \ne i\big\},$$

the training vectors falling into a particular region are approximated by a red dot associated with that region (Fig. 1).

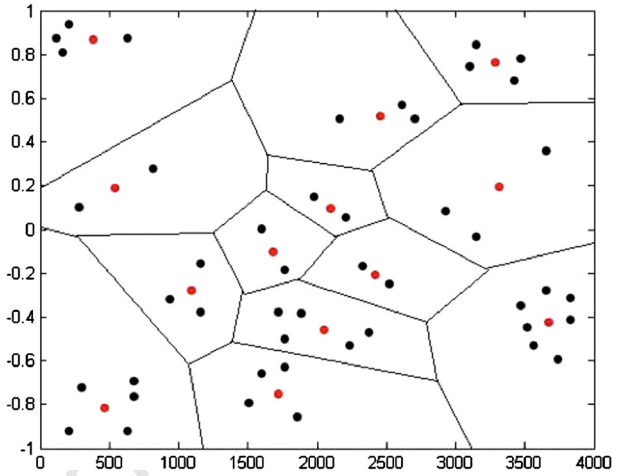

Fig. 1. Two-dimensional (2D) vector quantization. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

To find the optimal C and P, vector quantization uses a square-error distortion measure that specifies exactly how close the approximation is. The distortion measure is given as:

$$D_{ave} = \frac{1}{Mk} \sum_{m=1}^{M} \big\| X_m - Q(X_m) \big\|^2$$

If C and P are solution parameters to the minimization problem, then it must satisfy two conditions: (1) nearest-neighbor and (2) centroid. The nearest-neighbor condition indicates that the subregion S_n should consist of all the vectors that are closer to c_n than any of the other codevectors:

$$S_n = \big\{x\colon \|x - c_n\|^2 \le \|x - c_{n'}\|^2,\ \forall n' = 1, 2, \ldots, N\big\},$$

finally, the centroid condition indicates that the codevector c_n can be derived from the average of all the training vectors in its Voronoi region S_n:

$$c_n = \frac{\sum_{X_m \in S_n} X_m}{\sum_{X_m \in S_n} 1}, \quad n = 1, 2, \ldots, N.$$

As Elkan's discussed [34], the local learning techniques – use of c_n vectors for building a local model – prove that any intelligent learning model that examines all the attribute values of every training example must have the same or worse complexity. In other words, such a local learning strategy is far more efficient than that of the global learning strategy, especially on a large volume of data problems [37–40].

3.5. Deep learning

Shallow learning models (e.g., SVM, MLP, and GMM) have been widely used in the literature to solve simple or well-constrained problems. However, their limited modeling and representational power do not support their use in solving more complex problem, such as natural language problems. In 2006, the so-called deep learning (*a.k.a.* Representation learning) has emerged as new area of ML research [43–45] that exploits multiple layers of information-processing in a hierarchical architecture for pattern classification and or representation learning (e.g., Feed-forward neural networks) [46]. The main advantage of deep learning is referred to the drastically increased chip processing abilities, the lowered cost of computing hardware, and the recent advances in ML.

Please cite this article in press as: O.Y. Al-Jarrah et al., Efficient Machine Learning for Big Data: A Review, Big Data Research (2015), http://dx.doi.org/10.1016/j.bdr.2015.04.001

Deep neural networks (DNNs) are multilayer networks with many hidden layers, whose weights are fully connected and often initialized or pretrained using stacked Restricted Boltzmann Machine (RBM) or Deep Belief Networks (DBMs) [46]. DBM is a pretraining unsupervised step that utilizes large amount of unlabeled training data for extracting structures and regularities in input features [47]. DBN not only uses a huge amount of unlabeled training data but also provides good initialization weights for DNN. Moreover, overfitting and underfitting problems can be tackled by using the pretraining step of DBN. DNN has shown great performance in recognition and classification tasks, including natural language processing, image classification, and traffic flow detection [48]. However, DNN has high computational cost and difficult to scale [49]. DSN addresses the scalability problem of DNN, simple classifiers are stacked on top of each other in order to construct more complex classifier [50,51].

New techniques used in Sections 3.3 and 3.4 could fit to the problems of DNN naturally. The decision function of DNN is as follows:

$$P_j = \frac{\exp(x_j)}{\sum_k \exp(x_k)},$$

where P_j represents the class probability and x_j and x_k represent the total input to units j and k respectively. The cross entropy is defined as follows:

$$C = \sum_j d_j \log(p_j),$$

where d_j represents the target probability for output unit j, and P_j is the probability output for j after applying the activation function [52]. Now, the new semiparametric model's approximation is approximated as:

$$\frac{\exp(c_j)}{\sum_k \exp(c_k)} \approx \frac{\exp(x_j)}{\sum_k \exp(x_k)},$$

this approximation no longer extracts worst-case example x and is now able to reduce its complexity effectively. As in the local learning strategy, the model assumptions get stronger than those of nonparametric models, but they are less restrictive than those of parametric model while reducing its computational complexity significantly.

3.6. Big data computing

Big data computing systems fall into two major categories, based on how data is analyzed with regards to time constraint [53]. First, batch processing of large volumes of on-disk data with no time constraints (e.g., MapReduce and GraphLab). Second, streaming processing of in-memory data in real-time or short period of time (e.g., Storm, SAMOA) [54,55]. In [54], Huang and Liu argued that next-generation computing systems for big data analytics need innovative designs in both hardware and software that would provide a good match between big data algorithms and the underlying computing and storage resources.

There are several computing frameworks, e.g., Hadoop [56], SHadoop [57], ComMapReduce [58], Dryad [59], Piccolo [60], and IBM parallel machine learning toolbox, such systems have the capabilities to scale up machine learning. The combination of deep learning and parallel training implementation techniques provides potential ways to process Big Data [61]. Quoc V. Le et al. [62] consider the problem of building high-level, class-specific feature detectors from only unlabeled data. Experimental results reveal that it is possible to train a face detector without having to label images as containing a face or not.

K. Zhang and X. Chen [63] presented a distributed learning paradigm for the RBMs and the backpropagation algorithm using MapReduce. The DBNs are trained in a distributed way by stacking a series of distributed RBMs for pretraining and a distributed backpropagation for fine-tuning. Experimental results demonstrate that the distributed RBMs and DBNs are amenable to large-scale data with a good performance in terms of accuracy and efficiency.

4. Concluding remarks

In this review, we provided an overview of the current state of research in sustainable data modeling. In particular, we discussed its theoretical and experimental aspects in large-scale data-intensive fields, relating to: (1) model energy efficiency, including computational requirements in learning, and possible approaches, and (2) data-intensive areas' structure and design, including the relation between data models and characteristics. With the surge in e-science data, sustainable data modeling has been shown to offer a way forward due to its ease in handling large quantities of data. It is also envisaged that such data-modeling revolution can be readily extended to various areas in e-science. These newly designed sustainable data models will not only be able to cope with the emerging large-scale data paradigm, but also provide a means in maximizing its return for the various e-science areas.

Acknowledgements

The authors are grateful to Dr. Jason W.P. Ng at EBTIC, for his invaluable discussions and feedback, and special thanks to the British Telecom (BT) in London for their constructive criticism on this work.

References

[1] C. Pettey, Gartner estimates ict industry accounts for 2 percent of global co2 emissions, online, April 2007, http://www.gartner.com/newsroom/id/503867, accessed on Aug. 2014.
[2] S. Lohr, I.B.M. effort to focus on saving energy, online, May 2007, http://www.nytimes.com/2007/05/10/technology/10blue.html?pagewanted=print&_r=0.
[3] J.G. Koomey, et al., Estimating total power consumption by servers in the U.S. and the world, Technical report, Stanford University, 2007.
[4] M. Strathman, Model-based energy management reduces energy costs, E&P (2010), online, http://www.epmag.com/guests/2010/06/16/model-based-energy-management-reduces-energy-costs.
[5] J. Nil, Managing data ood is industry challenge, 2005.
[6] P. Yoo, J. Ng, A. Zomaya, An energy-efficient kernel framework for large-scale data modeling and classification, in: 2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), May 2011, pp. 404–408.
[7] H. Cheng, P.-N. Tan, R. Jin, Efficient algorithm for localized support vector machine, IEEE Trans. Knowl. Data Eng. 22 (4) (2010) 537–549.
[8] D.D.P.P. Lamb, R. Jurdak, Csiro ict centre and csiro sensors and sensor networks tcp, online, 2009, http://www.csiro.au/.
[9] M. Lautenschlager, Model and Data, Max-Planck-Institute for Meteorology, Hamburg, Germany.
[10] U.D. Energy, Insights learned from the human DNA sequence, what has been learned from analysis of the working draft sequence of the human genome? What is still unknown?, online, http://www.ornl.gov/hgmis, accessed on 2nd May 2011.
[11] A.J. Hey, A.E. Trefethen, The data deluge: an e-science perspective, 2003.
[12] NCBI, Genebank statistics, online, http://www.ncbi.nlm.nih.gov/genbank/genbankstats.html, accessed on 2nd Aug. 2014.
[13] Uniprotkb/swiss-prot protein knowledgebase release 2011_04 statistics, online, http://expasy.org/sprot/relnotes/relstat.html, accessed on 10th April 2011.
[14] Uniprotkb/trembl protein knowledgebase release 2011_04 statistics, online, http://www.ebi.ac.uk/uniprot/TrEMBLstats, accessed on 10th April 2011.
[15] T. Baluja, Electronic patient records will soon end doctors scrawl on paper, the globe and mail, online, http://www.theglobeandmail.com/news/national/toronto/electronic-patient-records-will-soon-end-doctors-scrawl-on-paper/article1982647, accessed on 13th April 2011.
[16] Insiteone official website, online, http://www.insiteone.com/, accessed on 10th April 2011.
[17] Nuclear Cardiology Markets, TriMark Publications, LLC, 2007.

[18] EMR, E. News, Dell launches new cloud-based services for hospitals and physician practices, online, http://www.emrandhipaa.com/news/2011/02/21/dell-launches-new-cloud-based-services-for-hospitals-and-physician-practice, accessed on 3rd April 2011.
[19] A.M. McPherson, et al., Vista: project status, in: SPIE 6267, Ground-based and Airborne Telescopes, 626707 (23 June 2006), vol. 6267, 2006, http://spie.org/Publications/Proceedings/Paper/10.1117/12.671352.
[20] The dark energy survey, online, http://www.darkenergysurvey.org/, accessed on 20th April 2011.
[21] The sloan digital sky survey, http://www.sdss.org/, accessed on 20th April 2011.
[22] N. Kaiser, et al., The pan-starrs survey telescope project, Bull. Am. Astron. Soc. 37 (2007) 1409.
[23] S. Ref, Lsst receives $30 million from Charles Simonyi and Bill Gates, online, http://www.spaceref.com/news/viewpr.html?pid=24409, accessed on 25th April 2011.
[24] M. Marwah, A. Shah, C. Bash, C. Patel, N. Ramakrishnan, Using data mining to help design sustainable products, Computer 44 (8) (2011) 103–106.
[25] N. Sundaravaradan, D. Patnaik, N. Ramakrishnan, M. Marwah, A. Shah, Discovering life cycle assessment trees from impact factor databases, in: AAAI, 2011.
[26] N. Sundaravaradan, M. Marwah, A. Shah, N. Ramakrishnan, Data mining approaches for life cycle assessment, in: IEEE ISSST, vol. 11, 2011.
[27] D. Patnaik, M. Marwah, R.K. Sharma, N. Ramakrishnan, Data mining for modeling chiller systems in data centers, in: Advances in Intelligent Data Analysis, vol. IX, Springer, 2010, pp. 125–136.
[28] P. Yang, Y. Hwa Yang, B.B. Zhou, A.Y. Zomaya, A review of ensemble methods in bioinformatics, Current Bioinform. 5 (4) (2010) 296–308.
[29] R. Maclin, D. Opitz, Popular ensemble methods: an empirical study, J. Artif. Intell. Res. (1999).
[30] R. Polikar, Ensemble based systems in decision making, IEEE Circuits Syst. Mag. 6 (3) (2006) 21–45.
[31] Y. Tang, Y. Wang, K. Cooper, L. Li, Towards big data Bayesian network learning – an ensemble learning based approach, in: 2014 IEEE International Congress on Big Data (BigData Congress), June 2014, pp. 355–357.
[32] Wikipedia, Ensemble in numerical weather prediction, online, http://en.wikipedia.org/wiki/Ensemble_forecasting, accessed on 23rd Feb. 2011.
[33] J.A.R. Blais, Estimation and Spectral Analysis, Univ. Calgary Press, Canada, 1988.
[34] C. Elkan, Naive Bayesian learning, Department of Computer Science and Engineering, University of California, San Diego, 1997.
[35] T. Joachims, Training linear svms in linear time, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2006, pp. 217–226.
[36] P.D. Yoo, Y.S. Ho, B.B. Zhou, A.Y. Zomaya, Siteseek: post-translational modification analysis using adaptive locality-effective kernel methods and new profiles, BMC Bioinform. 9 (1) (2008) 272.
[37] L. Bottou, V. Vapnik, Local learning algorithms, Neural Comput. 4 (6) (1992) 888–900.
[38] N. Gilardi, S. Bengio, Local machine learning models for spatial data analysis, J. Geogr. Inform. Dec. Anal. 4 (2000) 11–28, EPFL-ARTICLE-82651.
[39] H. Zhang, A.C. Berg, M. Maire, J. Malik, Svm-knn: discriminative nearest neighbor classification for visual category recognition, in: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, IEEE, 2006, pp. 2126–2136.
[40] K. Lau, Q. Wu, Local prediction of non-linear time series using support vector regression, Pattern Recognit. 41 (5) (2008) 1539–1547.
[41] P.D. Yoo, A.R. Sikder, J. Taheri, B.B. Zhou, A.Y. Zomaya, Domnet: protein domain boundary prediction using enhanced general regression network and new profiles, IEEE Trans. Nanobiosci. 7 (2) (2008) 172–181.
[42] A. Gersho, R.M. Gray, Vector Quantization and Signal Compression, Springer Science & Business Media, 1992.
[43] G. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.
[44] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[45] X.-W. Chen, X. Lin, Big data deep learning: challenges and perspectives, IEEE Access 2 (2014) 514–525.
[46] L. Deng, A tutorial survey of architectures, algorithms, and applications for deep learning, APSIPA Trans. Signal Inform. Process. 3 (2014), p. e2.
[47] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.
[48] Y. Lv, Y. Duan, W. Kang, Z. Li, F.-Y. Wang, Traffic flow prediction with big data: a deep learning approach, IEEE Trans. Intell. Transp. Syst. PP (99) (2014) 1–9.
[49] J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, Q.V. Le, A.Y. Ng, On optimization methods for deep learning, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp. 265–272.
[50] L. Deng, D. Yu, Deep convex net: a scalable architecture for speech pattern classification, in: Proceedings of the Interspeech, 2011.
[51] D.H. Wolpert, Stacked generalization, Neural Netw. 5 (2) (1992) 241–259.
[52] K.P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.
[53] A. Baldominos, E. Albacete, Y. Saez, P. Isasi, A scalable machine learning online service for big data real-time analysis, in: 2014 IEEE Symposium on Computational Intelligence in Big Data (CIBD), Dec. 2014, pp. 1–8.
[54] H. Huang, H. Liu, Big data machine learning and graph analytics: current state and future challenges, in: 2014 IEEE International Conference on Big Data (Big Data), Oct. 2014, pp. 16–17.
[55] A. Bifet, G.D.F. Morales, Big data stream learning with samoa, in: 2014 IEEE International Conference on Data Mining Workshop (ICDMW), Dec. 2014, pp. 1199–1202.
[56] Apache, Hadoop, http://hadoop.apache.org.
[57] R. Gu, X. Yang, J. Yan, Y. Sun, B. Wang, C. Yuan, Y. Huang, Shadoop: improving mapreduce performance by optimizing job execution mechanism in hadoop clusters, J. Parallel Distrib. Comput. 74 (3) (2014) 2166–2179.
[58] L. Ding, J. Xin, G. Wang, S. Huang, Commapreduce: an improvement of mapreduce with lightweight communication mechanisms, in: Database Systems for Advanced Applications, Springer, 2012, pp. 150–168.
[59] M. Isard, M. Budiu, Y. Yu, A. Birrell, D. Fetterly, Dryad: distributed data-parallel programs from sequential building blocks, in: ACM SIGOPS Operating Systems Review, vol. 41, ACM, 2007, pp. 59–72.
[60] R. Power, J. Li, Piccolo: building fast, distributed programs with partitioned tables, in: OSDI, vol. 10, 2010, pp. 1–14.
[61] C.P. Chen, C.Y. Zhang, Data-intensive applications, challenges, techniques and technologies: a survey on big data, Inf. Sci. 275 (2014) 314–347.
[62] Q.V. Le, Building high-level features using large scale unsupervised learning, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2013, pp. 8595–8598.
[63] K. Zhang, X. Chen, Large-scale deep belief nets with mapreduce, 2014.

Please cite this article in press as: O.Y. Al-Jarrah et al., Efficient Machine Learning for Big Data: A Review, Big Data Research (2015), http://dx.doi.org/10.1016/j.bdr.2015.04.001