Storages Are Not Forever

Erik Cambria[1] · **Anupam Chattopadhyay**[1] · **Eike Linn**[2] · **Bappaditya Mandal**[3] · **Bebo White**[4]

Keywords Information technology · Big Data analysis · Data storage · Data representation · Data learning · Data aggregation

Abstract Not unlike the concern over diminishing fossil fuel, information technology is bringing its own share of future worries. We chose to look closely into one concern in this paper, namely the limited amount of data storage. By a simple extrapolatory analysis, it is shown that we are on the way to exhaust our storage capacity in less than two cen-turies with current technology and no recycling. This can be taken as a note of caution to expand research initiative in several directions: firstly, bringing forth innovative data analysis techniques to represent, learn, and aggregate use-ful knowledge while filtering out noise from data; secondly, tap onto the interplay between storage and computing to minimize storage allocation; thirdly, explore ingenious solu-tions to expand storage capacity. Throughout this paper, we delve deeper into the state-of-the-art research and also put forth novel propositions in all of the abovementioned direc-tions, including space- and time-efficient data representa-tion, intelligent data aggregation, in-memory computing, extra-terrestrial storage, and data curation. The main aim of this paper is to raise awareness on the storage limitation we are about to face if current technology is adopted and the storage utilization growth rate persists. In the manuscript, we propose some storage solutions and a better utilization of storage capacity through a global DIKW hierarchy.

Introduction

The huge consumption of storage is a fact that is getting due attention within the umbrella term *Big Data*. While Big Data and their analysis bring in new research as well as com-mercial opportunities, the plain and simple fact behind this growth is that it is not there to last forever. A survey [1] shows that global consumption of storage in 2012 is 369 GB per capita, with top 4 countries contributing above 2 TB per capita. Digital universe is not only populated by individuals, for whom, one may argue that, the data is dispensable and hence, the storage crisis simply a storage recycling issue. This is partially true. In 2013, it was reported that over the course of diverse scientific experiments, CERN data cen-ter recorded over 100 PB (10^5 TB) data during previous 20 years [2].

[1] School of Computer Science and Engineering, Nanyang Technological University, 50 Nanyang Ave, Singapore 639798, Singapore

[2] IWE II, RWTH Aachen University, Aachen, Germany

[3] Institute for Infocomm Research, A*STAR, Singapore, Singapore

[4] SLAC National Laboratory, Stanford University, Stanford, CA, USA

The Panoramic Survey Telescope and Rapid Response System (Pan-STARRS) [3] generates 1.4 TB raw image data every night, which is used for impact risk assessment by NASA [4], apart from other scientific analysis, for detecting and avoiding collisions with near-Earth objects. The time horizon of such events, which is probed continuously, is 100 years. Large online companies regularly report data clusters in the orders of PB [5]. This data deluge is likely to continue with the emergence of new sensor-rich platforms.

These numbers, when put together in the context of the growth rate of storage consumption, lead to the capacity exhaustion scenario. To determine the limit of worldwide storage capacity, we begin with a set of available data about storage consumption as well as realistic assumptions as following.

Digital Storage Consumption Data

These are the data about digital storage consumption we leverage our analysis on:

- Worldwide storage size (s): 4.4 ZB (10^{21}) bytes [6].
- Worldwide storage growth rate (r): 40% per year [6].
- Atomic-scale storage is possible with magnetic memory [7]. With optimistic assumption, all atoms can be used. Total number of atoms (a): 10^{50} [8].

Digital Storage Assumptions

These are the following assumptions about digital storage we leverage our analysis on:

- Error correction techniques for recovering sensitive data will approach Shannon limit, though the increase in the data volume due to that is ignored for the present analysis.
- We are not recycling any storage at all. Admittedly, this is a strong assumption, which will be revisited later in section "In-memory Computing Technology" while considering in-memory computing.
- Data compression techniques are ignored for the first set of analyses provided in this section since it is inherently application-specific. We discuss that aspect in detail in section "Space-Efficient Data Representation."
- The current growth rate for the data usage will persist.

Storage Capacity Exhaustion

Based on the assumptions and available data, we deduce the time limit for storage capacity exhaustion.

Total number of bytes that can be stored considering all available atoms and using 12 atoms per bit [7] is

$$B = \frac{a}{12 \times 8} = \frac{10^{50}}{96} = 1.04 \times 10^{48}.$$

Assuming we have n number of years to exhaust this capacity, growing at a compound annual growth rate of r, we have the following relation.

$$(1+r)^n = \frac{1.04 \times 10^{48}}{s}$$
$$\implies 1.4^n = \frac{1.04 \times 10^{48}}{4.4 \times 10^{21}} = 2.36 \times 10^{26}.$$

Therefore,

$$n = \log_{1.4}\{2.36 \times 10^{26}\} = 180.47$$

Hence, we have roughly 181 years to exhaust all of the Earth's atomic storage capacity growing at the current rate of storage occupation.

Searching for Solutions

The rapid exhaustion of storage needs to be countered with scientific and technological innovations. In fact, there are various research directions that are being followed right now, which does not necessarily pay attention to this central issue.

The core motivation of this manuscript is to draw attention the fact that diminishing global storage capacity is a reality that needs focused research plan and visionary ideas.

Our contributions are primarily in relating the state-of-the-art research topics to the storage awareness. Further, we outline novel and futuristic research directions to address this concern. To discuss these in a systematic manner, in the following, we categorize the solution space into different segments with associated discussions.

Space-efficient data representation This stems from aggressive compression schemes and stretches towards statistical techniques to store the data in the most compact form, without incurring any loss of information. This is discussed in section "Space-Efficient Data Representation."

Time-efficient data interpretation This research addresses real-time learning paradigm, which, by the virtue of proper interpretation, can do away with any storage needs. We discuss this in section "Time-Efficient Data Learning."

Intelligent data aggregation The data, when acquired from an intelligent source, can be directly aggregated in a minimum necessary form that suffices the subsequent processing need. This approach is discussed in section "Structure-Efficient Data Aggregation."

In-memory computing architecture Emergence of recent non-volatile memory technologies, which can double up as computing devices, have opened up the scope of non-Von Neumann architectures. It can be also an extremely

viable architecture for addressing the storage crisis as discussed in section "In-memory Computing Technology."

Deep future storage We discuss the futuristic possibilities of storage capacity enhancement in section "The Future of Storage."

The above progression also represents the research focus from algorithm, software architecture, application model, and architecture model towards technology.

Space-Efficient Data Representation

When addressing volume in Big Data analytics, researchers in the data analytics community have largely taken a one-sided study of volume, which is the "Big Instance Size" factor of the data. The flip side of volume which is the dimensionality factor of Big Data, on the other hand, has received much less attention. The term "Big Dimensionality" [9] has been coined to put attention on the need for new ways in coping with the unprecedented number of features (dimensions) that are scaling to levels that now render existing approaches inadequate.

Dimensionality Reduction Techniques

Popular dimensionality reduction techniques include the following:

- *Missing values ratio:* Data columns with too many missing values are unlikely to carry much useful information. Thus, data columns with number of missing values greater than a given threshold can be removed. The higher the threshold, the more aggressive the reduction.
- *Low variance filter:* Similarly to the previous technique, data columns with little changes in the data carry little information. Thus, all data columns with variance lower than a given threshold are removed.
- *High correlation filter:* Data columns with very similar trends are also likely to carry very similar information. In this case, only one of them will suffice to feed the machine learning model. The correlation coefficient between numerical columns and between nominal columns can be calculated as the Pearson's product moment coefficient and the Pearson's chi-square value, respectively.
- *Random forests:* Decision tree ensembles, also referred to as random forests, are useful for feature selection in addition to being effective classifiers. One approach to dimensionality reduction is to generate a large and carefully constructed set of trees against a target attribute and then use each attribute's usage statistics to find the most informative subset of features.
- *Backward feature elimination:* In this technique, at a given iteration, the selected classification algorithm is trained on n input features. Then, one input feature is removed at a time and the same model is trained on $n - 1$ input features n times. The input feature whose removal has produced the smallest increase in the error rate is removed, leaving us with $n - 1$ input features. The classification is then repeated using $n - 2$ features, and so on. Each iteration k produces a model trained on $n - k$ features and an error rate $e(k)$. Selecting the maximum tolerable error rate, we define the smallest number of features necessary to reach that classification performance with the selected machine learning algorithm. Similar feature extraction algorithms for image classification are proposed in the literature [10].
- *Linear and non-linear component analysis:* In statistical analysis, for all supervised (labeled data) or unsupervised (unlabeled data) or semi-supervised (partially labeled data) methods, both linear and non-linear component analyses play a very big role for dimensionality reductions. Principal component analysis (PCA) [11] is a linear unsupervised statistical procedure that orthogonally transforms the original n coordinates of a data set into a new set of n coordinates called principal components. As a result of the transformation, the first principal component has the largest possible variance; each succeeding component has the highest possible variance under the constraint that it is orthogonal to (i.e., uncorrelated with) the preceding components. Keeping only the first $m < n$ components reduces the data dimensionality while retaining most of the data information, i.e., the variation in the data. While keeping the dimensions with largest energy (variance) certainly improves the density estimation, they are suboptimal for classification purposes. Linear discriminant analysis (LDA) is the most commonly used algorithm for both classification and dimensionality reduction. It attempts to minimize the Bayes error by selecting those feature vectors that maximize the ratio (Fisher criteria [11]) between the variance measure from the between class and variance measure from the within class. When our data can be modeled using a single Gaussian, although the underlying distributions of each class are not, PCA is preferred over LDA. Whereas when the classes correspond to linearly separable Gaussian distributions, LDA is generally preferred over PCA [12]. For very big data analysis, especially for their processing time and time of their availability, both PCA and LDA have their own limitations. To alleviate their limitations, researchers have proposed incremental PCA [13], incremental LDA [14], and many of their variants [15]. Their incremental procedures have improved the overall capability of handling large voluminous data but at

the expense of increased computational time. We take up a specific example to illustrate this discussion in the next section "Learn from Big Data and Then Delete It." In recent years, a significant amount of work is done in non-linear component analysis for dimensionality reduction. As PCA and LDA are popular approaches, they have been used as kernel or non-linear PCA [16] and kernel LDA [17]. The idea is to transform the data from original plane (in which the data is not linearly separable) to some other non-linear plane (very high-dimensional feature space), where the researchers envisage that the data could be linearly separable. Generally, polynomial, Gaussian, and radial basis functions are used for such transformation. In our opinion, on case-by-case basis, such transformation can be beneficial but not always. Moreover, such transformations are computationally expensive and involve a large number of free parameters. Comparisons on same databases and protocols using various component analyses, linear and non-linear, can be found in [18, 19].

PCA is possibly the most popular paradigm for dimensionality reduction. It needs to be noted that data compression is highly application driven, even making room for lossy and approximate value retention possibilities. Below, we describe one of the examples in details.

Learn from Big Data and Then Delete It

Large amount of data, e.g., video data, are predominantly applied for specific application purposes such as data reconstructions in communications or discriminant analysis for recognition. Once the knowledge is extracted from this vast amount of data, the original data are not required to enable or deploy these machine-driven applications. For example, Facebook's [20] DeepFace [21] used 4.4 million labeled faces from 4030 individuals (persons) of diverse face images to develop a deep learning architecture that helps in recognizing individuals. They have demonstrated the discriminative capability of such deep learning network by reducing the error of the current state-of-the-art by more than 25% on large unconstrained face and images (labeled faced in the wild) [22] and (YouTube) videos [23]. After all the learning, once they have the transformation basis feed-forward neural network (knowledge for distinguishing each individuals), they would no longer be required to keep the original data. Thereby, allowing us to perform specific task application without having the original enormous amount of data.

To further demonstrate this, we have collected 2388 images comprising of 1194 persons (two images FA/FB per person) selected from the FERET database [24]. Images are preprocessed following the CSU Face Identification Evaluation System [25]. Images are cropped into the size of 130×150. In this experiment, images of 250 people are randomly selected for training, and the remaining images of 944 people are used for testing, similar to [18]. There is no overlap in person between the training and testing sets. We test the popular PCA [26], fisher faces (FisherFace) [27], and eigenfeature regularization methodology (EigReg) proposed in [18, 28]. Cosine distance measure and the first nearest neighborhood classifier (1-NN) are applied to test all the machine learning approaches. At first, using the normalized face images of dimensionality $130 \times 150 = 19{,}500$ pixels (positive real numbers) and reducing the dimensions in the steps of 650 (randomly sampled) for 30 times, we compute the recognition rate as the percentage of the correct top 1 match on the testing set.

Figure 1 shows the recognition rate on the testing set against the number of features used in the matching. After applying various machine leaning techniques, the recognition rates against the number of features, varying each time in the steps of 10 for 24 times used in the matching process, are shown in Fig. 2. Figure 1 shows that using large number of features in the matching process, which inevitably also requires large storage space, the recognition rate is not as good as using very low number of features (for all the machine learning techniques), as shown in Fig. 2.

For a single point comparison, 10,000 positive real numbers are required to achieve an accuracy of around 87%, as shown in Fig. 1. To obtain similar accuracy using machine learning techniques, only 15 real numbers are required using EigReg method, as shown in Fig. 2. So there is a clear 99.85% improvement in the storage requirement for performing this face identification task. The above illustration shows that moving forward, we would need to develop critical learning mechanism so as to learn or extract knowledge (may be for specific purpose or a unified framework for general purpose) out of the enormous amount of data and then forget about the big data.

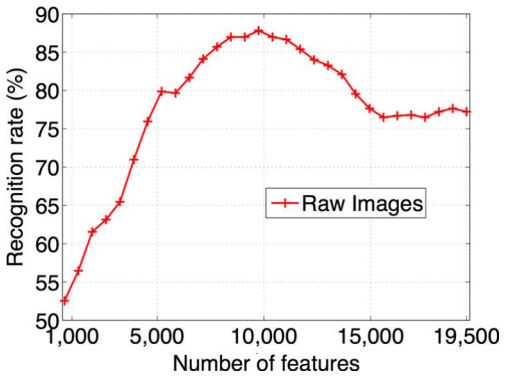

Fig. 1 Recognition rate with normalized images (without any training)

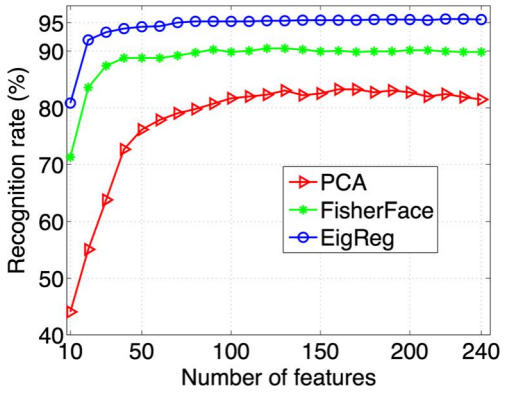

Fig. 2 Recognition rate after applying machine learning algorithms

Limitations and Roadmap

The problem with standard data compression is that it is not scalable: when the number of features and data points increases, the representation matrix becomes too high in dimensionality and too sparse for PCA to be calculated [29]. Some works have proposed fast approximations but they are only at most five times faster than standard PCA [30].

Lee et al. [31] conjectured that there might be simple but powerful meta-algorithms underlying neuronal learning that are fast, scalable, effective, and biologically plausible [29]. Optimizing all the $\approx 10^{15}$connections through the last few million years' evolution is very unlikely [29].

Instead, nature probably only optimizes the global connectivity, but leaves the other details to randomness [29]. In order to cope with the growing amount of data, we replace PCA with random projection (RP) [32], a method that maps the original dataset into a lower-dimensional subspace by using a Gaussian N(0, 1) matrix, while preserving the pair-wise distances with high probability (Fig. 3). This is possible thanks to the Johnson-Lindenstrauss (JL) Lemma [29], which states that with high probability, for all pairs of points x, y $\in$ X simultaneously,

$$\sqrt{\frac{m}{d}} \parallel x - y \parallel_2 (1-\varepsilon) \leq \parallel \Phi x - \Phi y \parallel_2 \leq \tag{1}$$

$$\leq \sqrt{\frac{m}{d}} \parallel x - y \parallel_2 (1+\varepsilon), \tag{2}$$

where X is a set of vectors in the Euclidean space, d is the original dimension of the space, m is the dimension of the smaller space, ε is a tolerance parameter measuring to what extent is the maximum distortion rate of the metric space, and Φ is a random matrix.

Structured RP for making matrix multiplication much faster was introduced in [33]. Achlioptas [34] proposed

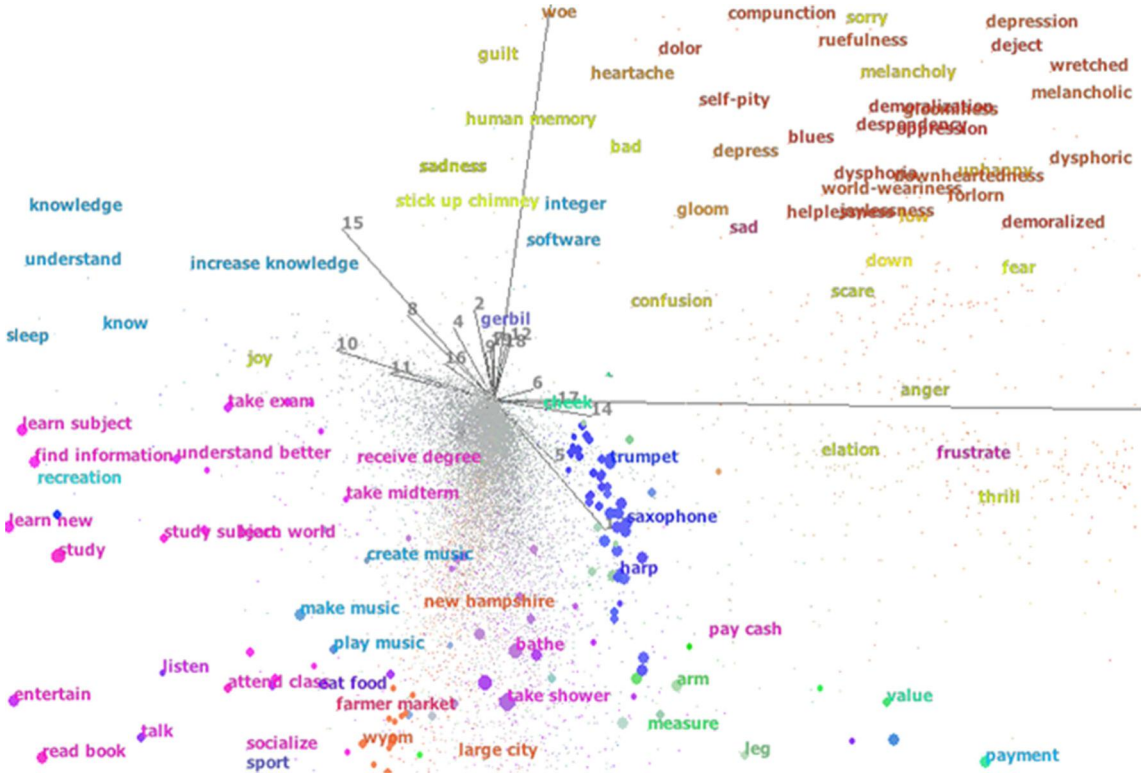

Fig. 3 Reduced (but equivalent) data representation

sparse RP to replace the Gaussian matrix with i.i.d. entries in

$$\phi_{ji} = \sqrt{s} \begin{cases} 1 & \text{with prob. } \frac{1}{2s} \\ 0 & \text{with prob.} 1 - \frac{1}{s} \\ -1 & \text{with prob.} \frac{1}{2s} \end{cases}, \tag{3}$$

where one can achieve a $\times 3$ speedup by setting $s = 3$, since only one third of the data need to be computed.

In applications where the data matrix is already quite sparse, however, it is not recommendable to use sparse RP. Instead, we propose to use subsampled randomized Hadamard transform (SRHT), as it behaves very much like Gaussian random matrices but accelerates the process from $\mathcal{O}(n\,d)$ to $\mathcal{O}(n\,log\,d)$ time [35].

Following [35, 36], for $d = 2^p$ where p is any positive integer, a SRHT can be defined as follows:

$$\Phi = \sqrt{\frac{d}{m}} \text{RHD} \tag{4}$$

where, • m is the number we want to subsample from d features randomly.

- R is a random $m \times d$ matrix. The rows of R are m uniform samples from the standard basis of $\mathbb{R}^d$.
- H$\in \mathbb{R}^{d\times d}$ is a normalized Walsh-Hadamard matrix, defined recursively $H_d = \begin{bmatrix} H_{d/2} & H_{d/2} \\ H_{d/2} & H_{d/2} \end{bmatrix}$ with $H_2 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix}$.
- D is a $d \times d$ diagonal matrix and the diagonal elements are i.i.d. Rademacher random variables.

Since the data analysis in this space only relies on the distances between vectors, it is possible to perform tasks such as categorical or analogical reasoning in a much reduced space with the same accuracy as the one of the original space.

Time-Efficient Data Learning

Randomness may not only be a possible solution for data representation but also for learning. Although fundamental in many areas of science, randomness is really native to computer science [37]. In the 1960s, its computational nature was clarified by Kolmogorov et al. [38] who proposed the first successful theory of random objects, defined roughly as those that cannot be computed from short descriptions.

Kolmogorov also suggested that randomness may have an important relationship with nondeterminism, namely, that the task of finding a "nonrandomness" witness (i.e., short fast program generating a given string) may be a good candidate to prove that exhaustive search cannot be avoided. An interesting proposition of learning automata is put forward by Kumpati et al. [39], which is an early representative of the large body of research in machine learning.

In the context of Big Data, randomness can be key to address emerging needs such as fast learning speed and big dimensionality reduction. When dealing with highly dynamic and highly dimensional data, minimal human intervention and efficient data representation are important factors for making sense of Big Data streams. Because of Big Data's volume, velocity, and variety, in fact, standard data representation techniques and learning methods are bound to fail. To this end, we propose the adoption of extreme learning machine (ELM) [40–42] (Fig. 4), an emerging technique that provides efficient unified solutions to generalized feedforward networks and, hence, has strong potential as a viable alternative technique for large-scale computing and machine learning in many different application fields, including big social data analysis [43] and commonsense reasoning [44]. ELM theory shows that the hidden neurons of generalized feedforward networks do not need to be tuned but instead just randomly generated, as their parameters are independent from the target functions or the training datasets.

ELM theories conjecture that this randomness may be true to biological learning in animal brains [45]. The ELM approach was introduced to overcome some issues in backpropagation network [46] training, specifically, potentially slow convergence rates, the critical tuning of optimization parameters [47], and the presence of local minima that call for multi-start and re-training strategies. The ELM learning problem settings require a training set, X, of N labeled pairs, where $(\mathbf{x}_i, y_i)$, where $\mathbf{x}_i \in \mathcal{R}^m$ is the ith input vector and $y_i \in \mathcal{R}$ is the associate expected "target" value; using a scalar output implies that the network has one output unit, without loss of generality.

The input layer has m neurons and connects to the "hidden" layer (having N_h neurons) through a set of weights $\{\hat{\mathbf{w}}_j \in \mathcal{R}^m;\ j = 1, ..., N_h\}$. The jth hidden neuron embeds

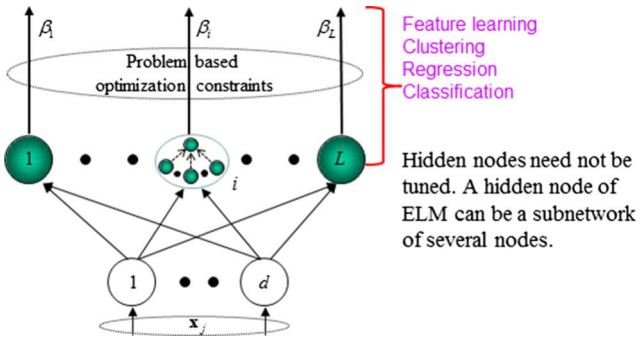

Fig. 4 Internal structure of an extreme learning machine (ELM)

a bias term, $\hat{b}_j$, and a nonlinear "activation" function, $\varphi(\cdot)$; thus, the neuron's response to an input stimulus, $\mathbf{x}$, is:

$$a_j(\mathbf{x}) = \varphi(\hat{\mathbf{w}}_j \cdot \mathbf{x} + \hat{b}_j) \tag{5}$$

Note that (5) can be further generalized to a wider class of functions [48] but for the subsequent analysis, this aspect is not relevant. A vector of weighted links, $\bar{\mathbf{w}}_j \in \mathcal{R}^{N_h}$, connects hidden neurons to the output neuron without any bias [45]. The overall output function, $f(\mathbf{x})$, of the network is

$$f(\mathbf{x}) = \sum_{j=1}^{N_h} \bar{\mathbf{w}}_j a_j(\mathbf{x}) \tag{6}$$

It is convenient to define an "activation matrix," $\mathbf{H}$, such that the entry $\{h_{ij} \in \mathbf{H}; i = 1, ..., N; j = 1, ..., N_h\}$ is the activation value of the jth hidden neuron for the ith input pattern. The $\mathbf{H}$ matrix is

$$\mathbf{H} \equiv \begin{bmatrix} \varphi(\hat{\mathbf{w}}_1 \cdot \mathbf{x}_1 + \hat{b}_1) & \cdots & \varphi(\hat{\mathbf{w}}_{N_h} \cdot \mathbf{x}_1 + \hat{b}_{N_h}) \\ \vdots & \ddots & \vdots \\ \varphi(\hat{\mathbf{w}}_1 \cdot \mathbf{x}_N + \hat{b}_1) & \cdots & \varphi(\hat{\mathbf{w}}_{N_h} \cdot \mathbf{x}_N + \hat{b}_{N_h}) \end{bmatrix} \tag{7}$$

In the ELM model, the quantities $\{\hat{\mathbf{w}}_j, \hat{b}_j\}$ in Eq. 5 are set randomly and are not subject to any adjustment, and the quantities $\{\bar{\mathbf{w}}_j, \bar{b}\}$ in Eq. 6 are the only degrees of freedom.

The training problem reduces to the minimization of the convex cost:

$$\min_{\{\bar{\mathbf{w}}, \bar{b}\}} ||\mathbf{H}\bar{\mathbf{w}} - \mathbf{y}||^2 \tag{8}$$

A matrix pseudo-inversion yields the unique L_2 solution, as proven in [49]:

$$\bar{\mathbf{w}} = \mathbf{H}^{+}\mathbf{y} \tag{9}$$

The simple, efficient procedure to train an ELM therefore involves the following steps:

1. Randomly set the input weights $\hat{\mathbf{w}}_i$ and bias $\hat{b}_i$ for each hidden neuron.
2. Compute the activation matrix, $\mathbf{H}$, as per (7).
3. Compute the output weights by solving a pseudo-inverse problem as per (9).

Despite the apparent simplicity of the ELM approach, the crucial result is that even random weights in the hidden layer endow a network with a notable representation ability [49]. Moreover, the theory derived in [50] proves that regularization strategies can further improve its generalization performance. As a result, the cost function (8) is augmented by an L_2 regularization factor as follows:

$$\min_{\bar{\mathbf{w}}} \{||\mathbf{H}\bar{\mathbf{w}} - \mathbf{y}||^2 + \lambda||\bar{\mathbf{w}}||^2\} \tag{10}$$

Popular learning techniques, e.g., neural networks and support vector machines, face some challenging issues such as intensive human intervene, slow learning speed, and poor learning scalability. It is clear that the learning speed of feedforward neural networks including deep learning is in general far slower than required and it has been a major bottleneck in their applications for past decades.

Two key reasons behind may be (1) the slow gradient-based learning algorithms are extensively used to train neural networks and (2) all the parameters of the networks are tuned iteratively by using such learning algorithms. ELM overcomes such issues by offering fast learning speed, ease of implementation, and minimal human intervention and, hence, represents a great solution for time-efficient data learning in Big Data environments.

Structure-Efficient Data Aggregation

Most important and unavoidable consequence of storage capacity exhaustion is the ability to analyze data and infer knowledge, which is expected to occupy smaller area compared to raw data. This is key especially in the context of the Social Web. Before 2003, there were just a few dozen exabytes of information on the World Wide Web. Today, that same amount of information is created weekly. The Web 2.0 has provided people with new services that allow them to create and share contents, ideas, and opinions, with virtually millions of other people connected to the Internet.

This huge amount of information, however, is specifically produced for human consumption and hence not directly processable by machines, as these are still very far from a minimum level of natural language understanding (NLU). So far, information retrieval has mainly been based on the textual representation of webpages and never really managed to fully to grasp the semantics of such text in an automatic way.

NLU, in fact, requires high-level symbolic capabilities, e.g., creation and propagation of dynamic bindings and manipulation of recursive constituent structures [51–53], which are necessary to shift from limited information retrieval techniques, e.g., counting word co-occurrence frequencies, to real NLU. Most of the current approaches to natural language processing are limited by the fact that they can only process the information that they can "see." The human brain, instead, can go far beyond that as every word encountered activates a cascade of semantically related concepts, relevant episodes, and sensory experiences.

Bridging the gap between the blind processing of text as bags of words and the human-like way to understand the meaning conveyed by natural language concepts will enable the transition from unstructured natural language data to structured machine-processable information and, hence, facilitate the transition to a global data-information-knowledge-wisdom (DIKW) hierarchy [54] (Fig. 5).

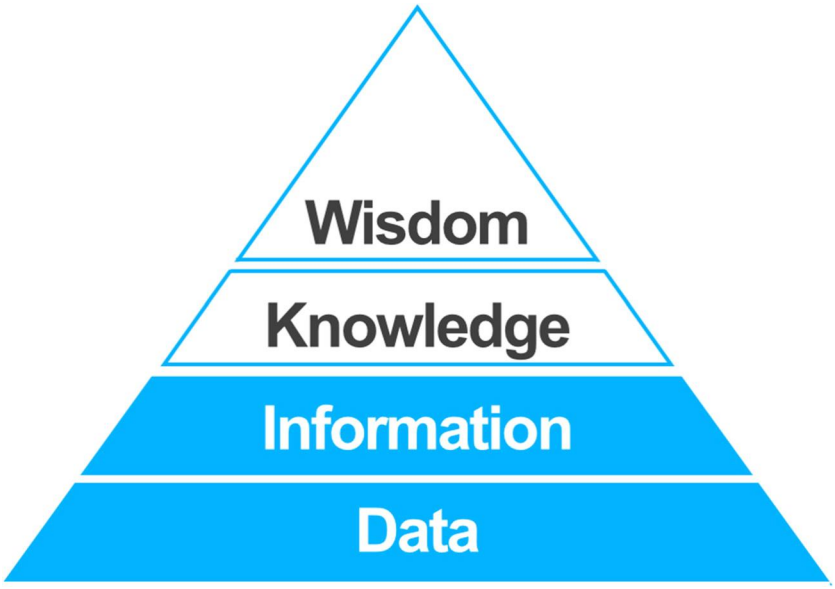

Fig. 5 Data-information-knowledge-wisdom (DIKW) hierarchy

In the context of big social data analysis [55], in particular, such a transition is enabled by sentic computing [56], a multidisciplinary approach to NLU that exploits an ensemble of machine learning [53], linguistics [57], and commonsense reasoning [58] to interpret and aggregate big social data. Sentic computing attempts to understand the underlying meaning of words and multiword expressions by means of commonsense knowledge and conceptual primitives (instead of counting word co-occurrence frequencies) and gives high importance to sentence structure (instead of treating text as bags of words).

An example of how sentic computing enables the transition from unstructured natural language data to structured machine-processable information is given by crowd validation [59], a process for comparing unstructured patient data with the structured healthcare ratings available for each hospital in the UK.

In particular, crowd validation deconstructs patients' stories into specific aspects or opinion targets (e.g., service, staff, timeliness) and polarity values associated with these (e.g., positive, negative, and neutral) in order for such stories to be more easily aggregated and compared with the official hospital ratings provided by the UK National Health Service (NHS).

In-memory Computing Technology

In the 1940s, John von Neumann introduced the stored-program computer where program and instruction data are stored in the same electronic memory. The basic principle is still valid for today's computer systems, but there is now a whole memory hierarchy: magnetic hard disc (magnetic), SSDs (Flash), working memory (DRAM), and L3 to L1 cache (SRAM). The main reason is that none of the available types of memory fulfills all the requirements:

- Non-volatile storage (good retention)
- Fast read/write access
- High cycle endurance
- Small size
- Small energy consumption
- Back-end of line (BEOL) compatibility
- Ultra dense array/3D stackability

A so-called universal memory would combine all the benefits from today's available memories. When having a look to the ITRS emerging research device (ERD) roadmap [60], there are listed several emerging memory technologies which could fulfill the abovementioned requirements. The best projected feature size there is in the range of 5 nm and the smallest cell area is $4F^2$ for resistively switching memories (ReRAM) [61].

For the conductive bridge-type ReRAM [62], a minimum feature size of 4 nm was calculated [63]. Those ReRAMs offer multi-levels, up to eight levels by now [64], i.e., $N = 3$ Bit, and enable 3D stacking of n layers in passive crossbar arrays [65]. If we assume $F = 5$ nm, $N = 3$ Bit and an $\frac{4F^2}{n}$ 3D array architecture, a theoretical storage density of $6 \cdot 10^{24}\text{Bit/m}^3$ is feasible. If we limit the available storage per person to $1m^3$ and assume $10 \cdot 10^9$ people in 2100, up to $6 \cdot 10^{33}$ Bit (7.510^{32} byte) will be available in total.

According to the previous calculation, we will face this limit already in 77 years, i.e., 2092. Note, if we assume enormous achievements in science, close-to-the-physical-limit kind of memories might be feasible at some point

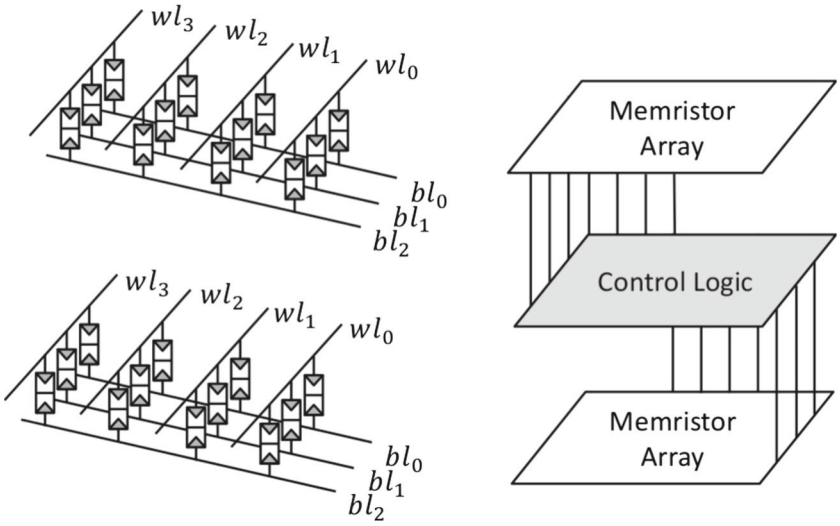

Fig. 6 In-memory computing architecture

in the future. The maximum possible storage density limit is $2.6 \cdot 10^{33}\text{Bit/m}^3$ [66]. Hence, another nine orders of magnitude in storage increase might be realizable. This would shift reaching the limit by about 60 years.

Interestingly, a universal memory will enable new ways to perform computing which might help to mitigate this problem. For example, ReRAMs or memristive devices [67, 68] have the ability to perform primitive Boolean logic [69, 70] as well as ternary arithmetic [71]. This raises the possibility of in-memory computing architectures, such the one shown in Fig. 6. Such a computing architecture with arrays of ReRAM and a control logic can switch between storage and computing mode. For example, the CRS logic approach [72] enables a highly efficient in-memory implementations of different arithmetic blocks [73–75] as well as other applications [76, 77]. Ultimately, also a so-called universal memcomputing machine [78] is envisioned, but practical realization using real memristsive devices is lacking up to now.

Memristive devices may induce a paradigm shift in computer architectures, dissolving the discrimination between logic gates and memory cells. The bad news is that a storage crisis will be equivalent to an information processing crisis in the future [79]. On the other hand, since the universal memory would cancel the separation of processing unit and storage, future architectures will eliminate the storage overhead in terms of storing the same piece of information at different memory hierarchy levels. Moreover, in such an architecture, even big amounts of data can be processed in real time, making temporary storage of incoming data unnecessary.

By filtering the interesting data out of the available data instantaneously, useless data will be directly discarded. Such an approach will hopefully help to reduce the storage growth rate drastically by storing only required and useful data in an intelligent way. Here, the analogy to the brain is helpful: we obtain a lot of data through our many sensors, but, we directly process these data and only store relevant data in our long-term memory.

Future machines with a certain degree of intelligence may be able to filter sensor data directly in the future, too. Thus, realizing more intelligent machines might help. Another point is the biological mechanism of forgetting which could find its way into future storage: Information of minor value which is not accessed for certain amount of time will automatically vanish and corresponding storage will be recycled. For example, some sorts of memristive devices offer a meta-stable behavior which could fulfill this property.

The Future of Storage

By revisiting the prediction of exhausting global storage, we can see that the scenario is a bit more optimistic, if we assume data recycling, peer-to-peer data sharing,[1] and the hope that extra-terrestrial storage capacity building is possible. Let us limit the study to the galactic storage.

From the observations based on luminosity and distribution of stellar mass, it is estimated that Milky Way galaxy has roughly 100 billion, i.e., 10^{11}, stars [80]. Assuming average stellar mass to be equivalent to the solar mass of 1.98×10^{33} g [81], we have a total stellar mass in our galaxy to be 1.98×10^{44} g. Each gram of matter contains an equivalent of 10^{24} hydrogen atoms, leading to the total galactic atom count to be 1.98×10^{68}. By repeating the above

[1] https://storj.io/

calculations with this count, we obtain the number of years for galactic storage to exhaust as following (denoted by n_g).

$$(1+r)^{n_g} = \frac{1.98 \times 10^{68}}{96 \times s}$$
$$\implies 1.4^{n_g} = \frac{2.06 \times 10^{66}}{4.4 \times 10^{21}} = 4.69 \times 10^{44}.$$

Therefore,

$$n_g = \log_{1.4}\{4.69 \times 10^{44}\} = 305.7.$$

By undertaking the impossibly hard scientific and technical challenge of converting every atom in the galaxy to a storage medium would extend the storage exhaustion deadline by 125 years. Taking fundamental physical principles into consideration, a universal form of entropy bound was proposed in [82], also known as the *Bekenstein bound*. An approximation of Bekenstein bound using mass-energy equivalence is

$$I = 2.577 \times 10^{43} mR, \tag{11}$$

where I is the information expressed in bits, m is the mass of the sphere expressed in kilograms, and R is the radius of the sphere in meters. Taking the mean radius of Earth and the mass, we obtain the information that can be contained in the entire Earth to be

$$I = 2.577 \times 10^{43} \times 5.972 \times 10^{24} \times 6.371 \times 10^{3}$$
$$= 9.81 \times 10^{71} bits$$

It indicates that the situation does not improve in terms of digital real estate, even if we lead to the absolute physical bounds of the storage.

Further and more importantly, even though one can move towards dense data storage, the information processing capacity does not scale arbitrarily either. This scaling is limited by energy, explored in [79] for quantum scale. Hence, though we focus on the individualistic goal of squeezing memory capacity, it challenges the corresponding computation rates as well, and hence, the limits are also affected by the operations that can be or need to be performed on the stored data.

Novel Storage Media: Extra-terrestrial Storage

It might be argued that the search of novel and denser storage can regain the balance in favor of everlasting, unlimited storage, which is, unfortunately, not the case. Though it is argued with practical demonstrations that biological storages fare better in terms of information density [83], it is also reported that 4.5×10^{20} bytes per gram is the theoretical maximum that can be achieved with single-stranded DNA [83]. This is not far from our optimistic assumptions of atomic storage. While the capacities after practical realization may differ significantly, it does not change the scenario for storage exhaustion.

In the quest to achieve interplanetary data communications, InterPlanetary Networking Special Interest Group (IPNSIG) is working towards a delay- and disruption-tolerant network (DTN) model that takes into account long/variable delay and high error rates, among others, to cope with the interplanetary communication issues. The underlying technology uses a principle based on *store-and-forward message switching* [84]. Since the communicating nodes often demonstrate intermittent connectivity, the s tore-and-forward technique allows treating delays and dis- r uptions in isolation. A key feature of this protocol is that t he intermediate storage places can hold messages indefi- n itely, i.e., the storage is permanent unlike the buffers in the i nternet routing protocol.

The dearth of terrestrial storage can be addressed by e xploiting extra-terrestrial storage and connecting those with InterPlaNetary (IPN) internet via the aforementioned DTN protocol. The preliminary requirement is, naturally, to be able to harvest a storage medium outside Earth and effectively converting it into a reliable storage via intelligent and controllable manufacturing technique. The steps of such a manufacturing is envisioned as the following.

1. Locate and identify a material for storage capacity.
2. Process and convert the material to a fully capable storage device.
3. Connect the storage device to a processing node.
4. Transfer and store the data.

Digital Curation

Related to the discussions of this manuscript, a related discipline that stemmed out of traditional archiving of library and museum is digital curation.

Current models of digital curation focuses more on long-term preservation of data, ease of access, format migration, and disposal [85]. Surprisingly, despite the influx of rich sensing platforms and Big Data, there is a lack of synergy between personal/enterprise-scale storage and digital curation. The research challenges outlined in the previous sections clearly show that by getting in close contact with the information aggregation, information processing, and even computing, digital curation can benefit. Indeed, there is a significant gap in storage management that can be addressed by *immersive and intelligent digital curation*. The goal of such curation practice could be the following:

- To perform real-time *immersive* curation by leveraging non-volatile on-chip memory
- To put the space-time-efficient data analytics and aggregation algorithms into practice and, thereby, strictly impose the DIKW hierarchy in digital curation

Conclusion

The world continues to generate quintillion bytes of data daily, leading to the pressing needs for new efforts in dealing with the grand challenges brought by Big Data. Today, there is a growing consensus that data volume presents an immediate challenge pertaining to the scalability issue.

In this manuscript, we discussed the eventual shortage of storage that is bound to happen under reasonable assumptions. Depending on the sets of assumptions, the storage capacity of the Earth may exhaust within merely two centuries. This can be a wake-up call to search for novel storage solutions, e.g., extra-terrestrial storage. In parallel, the representation, learning, and aggregation of data will have to be performed in a space-, time-, and structure-efficient manner, as well as considering human-like perceptions for knowledge discovery [86], in order to implement a global DIKW hierarchy that allows storage capacity to be better utilized.

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

Informed Consent Informed consent was not required as no human or animals were involved.

Human and Animal Rights This article does not contain any studies with human or animal subjects performed by any of the authors.

References

1. Where in the world is storage: a look at byte density across the globe. www.idc.com/downloads/where_is_storage_infographic_243338.pdf. Accessed 08 June 2015.
2. CERN Data Center. http://home.web.cern.ch/about/updates/2013/02/cern-data-centre-passes-100-petabytes. Accessed 08 June 2015.
3. When the meteor and the 1PB database collide. http://www.computerworld.com/article/2532280/data-center/when-the-meteor-and-the-1pb-database-collide.html Accessed 08 June 2015.
4. NASA Near Earth Object Program. http://neo.jpl.nasa.gov/risks/doc/sentry_faq.html. Accessed 08 June 2015.
5. Moving an elephant: large scale Hadoop data migration at Facebook. https://www.facebook.com/notes/paul-yang/moving-an-elephant-large-scale-hadoop-data-migration-at-facebook/10150246275318920. Accessed 08 June 2015.
6. The digital universe of opportunities: rich data and the increasing value of the internet of things. http://www.emc.com/leadership/digital-universe/2014iview/executive-summary.htm. Accessed 08 June 2015.
7. Loth S, Baumann S, Lutz CP, Eigler DM, Heinrich AJ. Bistability in atomic-scale antiferromagnets. Science. 2012;335(6065):196–9.
8. Physics questions people ask Fermilab. http://www.fnal.gov/pub/science/inquiring/questions/atoms.html. Accessed 08 June 2015.
9. Zhai Y, Ong Y-S, Tsang I. The emerging "big dimensionality". IEEE Comput Intell Mag. 2014;9(3):14–26.
10. Haralickand R, Shanmugam K, Dinstein I. Textural features for image classification. IEEE Trans Syst Man Cybern. 2007;3(6):610–21.
11. Duda RO, Hart PE, Stork DG. Pattern classification. New York: Wiley; 2001.
12. Zhu M, Martinez AM. Optimal subclass discovery for discriminant analysis. In: Conference on computer vision and pattern recognition workshop, 2004. CVPRW '04. 2004. p. 97–104.
13. Wang M, Li H-X, Chen X, Chen Y. Deep learning-based model reduction for distributed parameter systems. IEEE Trans Syst Man Cybern Syst. 2016;46(12):1664–74.
14. Dai B, Li H, Wei L. Image processing unit for general-purpose representation and association system for recognizing low-resolution digits with visual information variability. IEEE Trans Syst Man Cybern Syst. 2016.
15. Zhao H, Yuen PC. Incremental linear discriminant analysis for face recognition. IEEE Trans Syst Man Cybern Part B (Cybern). 2008;38(1):210–21.
16. Schölkopf B, Mika S, Burges C, Knirsch P, Müller K-R, Rätsch G, Smola A. Input space versus feature space in kernel-based methods. IEEE Trans Neural Netw. 1999;10:1000–17.
17. Mika S, Ratsch G, Weston J, Scholkopf B, Mullers KR. Fisher discriminant analysis with kernels. In: Proceedings of the 1999 IEEE signal processing society workshop neural networks for signal processing IX. 1999. p. 41–48.
18. Jiang XD, Mandal B, Kot A. Eigenfeature regularization and extraction in face recognition. IEEE Trans Pattern Anal Mach Intell. 2008;30(3):383–94.
19. Jiang XD, Mandal B, Kot A. Complete discriminant evaluation and feature extraction in kernel space for face recognition. Mach Vis Appl Springer. 2009;20(1):35–46.
20. Facebook. Online social network. https://www.facebook.com/. 2015.
21. Taigman Y, Yang M, Ranzato M, Wolf L. Deepface: closing the gap to human-level performance in face verification. In: CVPR. Columbus; 2014. p. 1701–1708.
22. Huang GB, Ramesh M, Berg Ta, Learned-Miller E. Labeled faces in the wild: a database for studying face recognition in unconstrained environments. Technical Report 07-49. University of Massachusetts, Amherst. 2007.
23. Wolf L, Hassner T, Maoz I. Face recognition in unconstrained video with matched background similarity. In: IEEE Conference on computer vision and pattern recognition. 2011. p. 529–534.
24. Phillips PJ, Moon H, Rizvi S, Rauss P. The FERET evaluation methodology for face recognition algorithms. IEEE Trans Pattern Anal Mach Intell. 2000;22(10):1090–1104.
25. The Face Recognition Technology (FERET) Normalization. http://www.cs.colostate.edu/evalfacerec/data/normalization.html. CSU.
26. Turk M, Pentland A. Eigenfaces for recognition. J Cogn Neurosci. 1991;3(1):71–86.
27. Swets DL, Weng J. Using discriminant eigenfeatures for image retrieval. IEEE Trans Pattern Anal Mach Intell. 1996;18(8):831–6.
28. Mandal B, Zhikai W, Li L, Kassim A. Whole space subclass discriminant analysis for face recognition. In: IEEE International conference on image processing (ICIP). Quebec City.
29. Balduzzi D. Randomized co-training: from cortical neurons to machine learning and back again. arXiv:1310.6536. 2013.
30. Menon AK, Elkan C. Fast algorithms for approximating the singular value decomposition. ACM Trans Knowl Discov Data (TKDD). 2011;5(2):13.
31. Lee H, Grosse R, Ranganath R, Ng AY. Unsupervised learning of hierarchical representations with convolutional deep belief networks. Commun ACM. 2011;54(10):95–103.

32. Bingham E, Mannila H. Random projection in dimensionality reduction: applications to image and text data. In: ACM SIGKDD. 2001. p. 245–250.
33. Sarlos T. Improved approximation algorithms for large matrices via random projections. In: FOCS. 2006. p. 143–152.
34. Achlioptas D. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. J Comput Syst Sci. 2003;66 (4):671–687.
35. Yichao L, Dhillon P, Foster DP, Ungar L. Faster ridge regression via the subsampled randomized hadamard transform. In: Advances in neural information processing systems. 2013. p. 369–377.
36. Tropp JA. Improved analysis of the subsampled randomized hadamard transform. Adv Adapt Data Anal. 2011;3(01n02):115–26.
37. Lewis L. Randomness and nondeterminism. In: International congress of mathematicians. Zurich. 1994.
38. Kolmogorov A, Uspenskii V. Algorithms and randomness. Theor Veroyatnost i Primenen. 1987;3(32):389–412.
39. Jiao L, Denoeux T, Pan Q. A hybrid belief rule-based classification system based on uncertain training data and expert knowledge. IEEE Trans Syst Man Cybern Syst. 2016;46(12):1711–23.
40. Cambria E, Huang G-B, et al. Extreme learning machines. IEEE Intell Syst. 2013;28(6):30–59.
41. Huang G-B, Zhu Q-Y, Siew C-K. Extreme learning machine: theory and applications. Neurocomputing. 2006;70(1):489–501.
42. Huang G-B, Cambria E, Toh K-A, Widrow B, Zongben X. New trends of learning in computational intelligence. IEEE Comput Intell Mag. 2015;10(2):16–7.
43. Oneto L, Bisio F, Cambria E, Anguita D. Statistical learning theory and ELM for big social data analysis. IEEE Comput Intell Mag. 2016;11(3):45–55.
44. Oneto L, Bisio F, Cambria E, Anguita D. Semi-supervised learning for affective common-sense reasoning. Cogn Comput. 2017;9(1):18–42.
45. Huang G-B. An insight into extreme learning machines: random neurons, random features and kernels. Cogn Comput. 2014;6(3):376–90.
46. Ridella S, Rovetta S, Zunino R. Circular backpropagation networks for classification. IEEE Trans Neural Netw. 1997;8(1):84–97.
47. Vogl TP, Mangis JK, Rigler AK, Zink WT, Alkon DL. Accelerating the convergence of the back-propagation method. Biol Cybern. 1988;59(4-5):257–63.
48. Huang G-B, Chen L, Siew C-K. Universal approximation using incremental constructive feedforward networks with random hidden nodes. IEEE Trans Neural Netw. 2006;17(4):879–92.
49. Huang G-B, Wang DH, Lan Y. Extreme learning machines: a survey. Int J Mach Learn Cybern. 2011;2(2):107–122.
50. Huang G-B, Zhou H, Ding X, Zhang R. Extreme learning machine for regression and multiclass classification. IEEE Trans Syst Man Cybern Part B: Cybern. 2012;42(2):513–29.
51. Dyer M. Connectionist natural language processing: a status report, volume 292 of Computational architectures integrating neural and symbolic processes. Dordrecht: Kluwer Academic; 1995, pp. 389–429.
52. Cambria E, White B. Jumping NLP curves: a review of natural language processing research. IEEE Comput Intell Mag. 2014; 9(2):48–57.
53. Chaturvedi I, Ong Y-S, Tsang IW, Welsch RE, Cambria E. Learning word dependencies in text by means of a deep recurrent belief network. Knowl-Based Syst. 2016;108:144–54.
54. Rowley J. The wisdom hierarchy: representations of the DIKW hierarchy. J Inf Sci. 2007;33(2):163–180.
55. Cambria E, Wang H, White B. Guest editorial: big social data analysis. Knowl-Based Syst. 2014;69:1–2.
56. Cambria E, Hussain A. Sentic computing: a common-sense-based framework for concept-level sentiment analysis. Cham: Springer; 2015.
57. Poria S, Cambria E, Gelbukh A, Bisio F, Hussain A. Sentiment data flow analysis by means of dynamic linguistic patterns. IEEE Comput Intell Mag. 2015;10(4):26–36.
58. Cambria E, Poria S, Bajpai R, Schuller B. SenticNet 4: a semantic resource for sentiment analysis based on conceptual primitives. In: COLING. 2016. p. 2666–2677.
59. Cambria E, Hussain A, Havasi C, Eckl C, Munro J. Towards crowd validation of the UK national health service. In: WebSci. Raleigh. 2010.
60. The international technology roadmap for semiconductors (ITRS). International technology roadmap for semiconductors - 2013 edition. http://dx.doi.org/http://www.itrs.net. 2013.
61. Menzel S, Linn E, Waser R. Redox-based resistive memory. Wiley; 2015. vol. 1, chapter 8, p. 137–161.
62. Valov I, Tappertzhofen S, Linn E, Menzel S, van den Hurk J, Waser R. Atomic scale and interface interactions in redox-based resistive switching memories. ECS Trans. 2014;64(14):3–18.
63. Zhirnov VV, Meade R, Cavin RK, Sandhu G. Scaling limits of resistive memories. Nanotechnology. 2011;22(25):254027/1–21.
64. Chien W-C, Lee M-H, Lee F-M, Lin Y-Y, Lung H-L, Hsieh K-Y, Lu C-Y. A multi-level 40nm WOX resistive memory with excellent reliability. In: 2011 IEEE international electron devices meeting IEDM '11. 2011.
65. Kügeler C, Meier M, Rosezin R, Gilles S, Waser R. High density 3D memory architecture based on the resistive switching effect. Solid State Electron. 2009;53(12):1287–92.
66. Lloyd S. Ultimate physical limits to computation. Nature. 2000;406:1047–54.
67. Strukov DB, Snider GS, Stewart DR, Williams RS. The missing memristor found. Nature. 2008;453(7191):80–3.
68. Chua LO, Kang SM. Memristive devices and systems. Proc IEEE. 1976;64(2):209–23.
69. Borghetti J, Snider GS, Kuekes PJ, Yang JJ, Stewart DR, Williams RS. 'Memristive' switches enable 'stateful' logic operations via material implication. Nature. 2010;464(7290):873–76.
70. Linn E, Rosezin R, Tappertzhofen S, Böttger U, Waser R. Beyond von Neumann—logic operations in passive crossbar arrays alongside memory operations. Nanotechnology. 2012;23: 305205.
71. Kim W, Chattopadhyay A, Siemon A, Linn E, Waser R, Rana V. Multistate memristive tantalum oxide devices for ternary arithmetic. Sci Rep. 2016;6:36652 EP –, 11.
72. Siemon A, Breuer T, Aslam N, Ferch S, Kim W, van den Hurk J, Rana V, Hoffmann-Eifert S, Waser R, Menzel S, Linn E. Realization of Boolean logic functionality using redox-based memristive devices. Adv Funct Mater. 2015.
73. Siemon A, Menzel S, Chattopadhyay A, Waser R, Linn E. In-memory adder functionality in 1S1R arrays. In: Proceedings of 2014 IEEE international symposium on circuits and systems (ISCAS). 2015.
74. Siemon A, Menzel S, Waser R, Linn E. A complementary resistive switch-based crossbar array adder. IEEE J Emerg Sel Top Circ Syst. 2015;5(1):64–74.
75. Breuer T, Siemon A, Linn E, Menzel S, Waser R, Rana V. A HfO2-based complementary switching crossbar adder. Adv Electron Mater. 2015.
76. Bhattacharjee D, Chattopadhyay A. Efficient binary basic linear algebra operations on reram crossbar arrays. In: 2017 30th

international conference on VLSI design and 2017 16th international conference on embedded systems (VLSID). 2017. p. 277–282.
77. Bhattacharjee D, Chattopadhyay A. In-memory data compression using ReRAMs. Springer International Publishing; 2017. p. 275–291.
78. Traversa FL, Di Ventra M. Universal memcomputing machines. IEEE Trans Neural Netw Learn Syst. [published online; doi:10.1109/TNNLS.2015.2]. 2015. p. 1–14.
79. Lloyd S. Ultimate physical limits to computation. Nature. 2000;406:1047–54.
80. How many stars are there in our galaxy (Milky Way)? http://curious.astro.cornell.edu/about-us/78-the-universe/stars-and-star-clusters/general-questions/343-how-many-stars-are-there-in-our-galaxy-milky-way-intermediate Accessed 09 June 2015.
81. 2014 Astronomical Constants. http://asa.usno.navy.mil/static/files/2014/Astronomical_Constants_2014.pdf. Accessed: 09 June 2015.
82. Bekenstein JD. Universal upper bound on the entropy-to-energy ratio for bounded systems. Phys Rev D. 1981;23(2):287–98.
83. Church GM, Gao Y, Kosuri S. Next-generation digital information storage in DNA. Science. 2012;337(6102):1628.
84. Delay- and disruption-tolerant networks (DTNs): a tutorial, version 2.0. http://ipnsig.org/links-for-academics-and-technical-folks/. Accessed 09 June 2015.
85. Higgins S. The DCC curation lifecycle model. Int J Digit Curat. 2008;3(1):134–40.
86. Klein G, Calderwood R, MacGregor D. Critical decision method for eliciting knowledge. IEEE Trans Syst Man Cybern. 2002;19(3):462–72.