Journal of Medical Systems (2018) 42: 139
https://doi.org/10.1007/s10916-018-1003-9

A Survey of Data Mining and Deep Learning in Bioinformatics

Kun Lan [1] · Dan-tong Wang [2] · Simon Fong [1] · Lian-sheng Liu [3] · Kelvin K. L. Wong [4] · Nilanjan Dey [5]

Received: 28 March 2018 / Accepted: 21 June 2018 / Published online: 28 June 2018

Abstract

The fields of medicine science and health informatics have made great progress recently and have led to in-depth analytics that is demanded by generation, collection and accumulation of massive data. Meanwhile, we are entering a new period where novel technologies are starting to analyze and explore knowledge from tremendous amount of data, bringing limitless potential for information growth. One fact that cannot be ignored is that the techniques of machine learning and deep learning applications play a more significant role in the success of bioinformatics exploration from biological data point of view, and a linkage is emphasized and established to bridge these two data analytics techniques and bioinformatics in both industry and academia. This survey concentrates on the review of recent researches using data mining and deep learning approaches for analyzing the specific domain knowledge of bioinformatics. The authors give a brief but pithy summarization of numerous data mining algorithms used for preprocessing, classification and clustering as well as various optimized neural network architectures in deep learning methods, and their advantages and disadvantages in the practical applications are also discussed and compared in terms of their industrial usage. It is believed that in this review paper, valuable insights are provided for those who are dedicated to start using data analytics methods in bioinformatics.

Keywords Bioinformatics · Biomedicine · Data mining · Machine learning · Deep learning

Introduction

Revolutionary changes and improvements have been witnessed in the research areas of biomedicine during the several past decades. As well known, biomedicine is a frontier and interdisciplinary subject derived from the theories and methodologies of comprehensive medicine, life science and biology. The basic task is to apply biology and engineering techniques to study and solve the problems in life science, especially in medicine [1, 2]. Biomedicine is the base of academic research and innovation of biomedical information, medical imaging technology, gene chip, nanotechnology,

This article is part of the Topical Collection on *Image & Signal Processing*

✉ Simon Fong
ccfong@umac.mo

Kun Lan
yb67405@umac.mo

Dan-tong Wang
wdt395@mail.zrtg.com

Lian-sheng Liu
llsjnu@sina.com

Kelvin K. L. Wong
Kelvin.Wong@westernsydney.edu.au

Nilanjan Dey
neelanjan.dey@gmail.com

[1] Department of Computer and Information Science, University of Macau, Taipa, Macau, China

[2] Department of Media integration technology center, Zhejiang Radio & TV Group, Hangzhou, People's Republic of China

[3] First Affiliated Hospital of Guangzhou University of TCM, Guangzhou 510405, Guangdong, China

[4] School of Medicine, University of Western Sydney, Sydney, NSW, Australia

[5] Department of Information Technology, Techno India College of Technology, Kolkata, West Bengal 740000, India

new material and so on. With the evolution of the social-psycho-biomedical model, the development of systems biology has formed the modern system of biomedicine. It is closely related to formation and biotechnology industry in the twenty-first century, and is an important engineering field associated with improving the level of medical diagnosis and human health [3–8]. And for bioinformatics, a new promising discipline which emerges and develops rapidly because of the contribution of biotechnology and data analysis methods, is probably regarded as a considerable component of traditional human health informatics for the interdisciplinary combination of biomedical information and computer science. The complete works and jobs done in bioinformatics show that from all kinds of aspects or views, great importance of bioinformatics is obtained and acquired to effectively analyze and extract valuable knowledge from increasingly massive biomedical information.

On the other hand, since the fast development of biotechnology has been accumulated within the historical period of time, the exponential increase rate of biomedical data generated by various research and application areas can range from micro molecular level (gene functions, protein interactions, etc.), biological tissue level (brain connectivity map, Magnetic resonance images, etc.), clinical patient level (intensive care unit, electronic medical record, etc.) and whole population level (medical message board, social media, etc.) [9–11]. The unneglectable fact is that growth speed and heterogeneous structure make it much more challenging to handle biomedical data with such properties than conventional data analysis methods as usual [12]. Therefore, there is a desirable need to create more powerful theoretical methodologies and practical tools for analyzing and extracting meaningful information from above mentioned complex bio-data. The key points behind those numerous efficient and scalable methods and tools include but not limited to classification, prediction, clustering, outlier detection, sequential processing, frequent item query, deep network architecture construction, spatial/temporal data analysis and visualization. To transfer useful knowledge from original raw data, data mining approach with core methodology called machine learning is proved to be a successful way and has been widely applied by building models, making predictions, doing classifications and clustering, finding associated rules, and finally uncovering wanted patterns. Meanwhile, deep learning is a more recent concept and framework, and has much better ability of feature representation in abstract level than general machine learning.

The problem appears under the circumstance of how to combine advanced data analysis methods and bioinformatics together for the purpose of bridging the two fields systematically and mining biological data successfully. As far as the contribution of this paper is concerned, the general overview is presented with sophisticated data analysis methods represented by data mining and deep learning techniques that have been utilized in the wide range of bioinformatics.

Data mining in bioinformatics

Behind the incremental datasets there are many important things, which require large capacity data storage devices and high capacity analysis tools. From a technical point of view, machine learning or data mining approach is also very necessary. It is features of abstract data representations that guarantee data mining can achieve the goal of accurate and reliable performance. However, human analysis and abstraction are not suitable for massive data with both high dimensional attributes and tremendous number of instances. In addition, the growth rate of data is much faster than that of the traditional manual analysis technology. And when we do not have the ability to translate the information into a more understandable representation to provide users with original data, the meaning of existence is also lost. So in order to make better use of this type of data to help clinical diagnosis and determine the clinical effects of drugs on experimental data, it is urgent to provide an automatic data analysis method for analyzing the high-level data. With the years of exploring and progressing in bioinformatics, there are many machine learning tools used for data investigation and analysis nowadays [13]. Data mining is involved in many biomedical study areas, such as biological electronics and nervous system, bioinformatics and computational biology, biological materials, biomedical imaging, image processing and visualization, biomedical modeling, gene engineering, medical cell biology, nano biological analysis, nuclear magnetic resonance/CT/ECG, physiological signal processing, etc.

Data preprocessing

Data preprocessing is a procedure dealing with the data before the main process. In the real world, some kinds of data are generally raw, noisy, incomplete and inconsistent, and cannot be directly used for data mining, or even their subsequent mining results are unsatisfactory. In order to enhance data mining performance, a preprocessing step is introduced as an important step in the whole process. It usually contains following methods: data cleaning, integration, transformation, reduction and so on [13]. These preprocessing techniques are applied before data mining, which greatly improves the quality of patterns and reduces the time required for actual mining.

Data cleaning

Data cleaning routines clean up data by filling in missing values, smoothing noise, identifying or deleting outliers, and resolving inconsistencies. Generally speaking, a set of criteria are introduced to evaluate the quality of data itself, they are validity (confirmation of whether each data is consistent with the original schema, such as data type, missing value, unique field, range, regular expression, constraints...), accuracy(the value indicated by the data is reasonable and correct, requires verification via other data), completeness (needs to verify through other data), consistency (there is an association between data fields about whether they are consistent with each other, and no contradiction occurs) and uniformity(the agreement of measuring unit used in the same field for each piece of data). After that, a number of actions are token to perform the cleaning produce such as data auditing, workflow specification, execution of plan and manually post-correction. Brazma, Alvis, et al. established a standard for recording and reporting microarray-based gene expression data, which is Minimum Information About a Microarray Experiment (MIAME) that format standardization, abnormal data removal, error correction, duplicate data removal [14]. Antonie, Maria-Luiza, et al. investigated the use of neural networks (NNs) and association rule mining, for anomaly detection and classification, which automatically sweeps through the medical image and cutting horizontally and vertically the image into those parts with the mean less than a certain threshold [15].

Data integration

Data integration routines combine data from multiple sources (flat files, databases, data cubes, data warehouses, etc.), store them together and provide users with a unified view of data. The process of building a data warehouse is in fact equal to data integration. Such works have been achieved by several researchers. Five common and initial types of integration methods for combining data into the valuable format, namely data consolidation (collecting various data into one consolidated storage), propagation (copying data between the source and destination synchronously or asynchronously), virtualization (a unified view is shown through real-time interface from multiple data models), federation (a virtual database on heterogeneous data sources) and warehouse (a storage repository for disparate data). The benefits offered by data integrations make pieces of anecdotal data together and give the users actionable insights and informative perspectives on data preprocessing. Dasu, Tamraparni, et al. developed such a system of high efficiency, Bellman that performs data integration on the structural database [16]. Raman, Vijayshankar, and Joseph M. Hellerstein presented a quick identifying similar values and estimating join directions and sizes [17]. Becker, Barry, et al. gave a simple Bayesian classifier (SBC) based on a conditional independence model for attribute processing [18]. In Zhang J., et al. and Xu, Xiaowei, et al.'s work, a clustering data processing approach called DClust that provides multi-resolution view of the clusters and generates arbitrary shapes clusters in the presence of noise [19, 20].

Data transformation

Data can be transformed into a specific form that is suitable for mining through smooth aggregation, data generalization and normalization. The nature of transformation is the replacement of latent pattern distribution shape or relationship for the purpose of computational convenience. Besides, easy but comprehensive visualization can be obtained via normalization of different scales of data in transformation step, and high interpretability is enhanced by aggregation and generalization operations on hierarchical data concepts, which enables the data usable and facilitates the efficiencies of data storage, management and computation. The entire workflow of data transformation involves the data discovery (understand the form of data that needs to be transformed), data mapping (it determines the rules of how data is extracted, filtered, revised and merged during the conversion),data execution (applying technical tools to implement the transformation) and data review (checking whether the output achieves the requirements).Han, Jiawei, et al. proposed a seven-step model that consists of many machine learning methods to implement data cleaning and preprocessing, data reduction and transformation [21]. Daubechies, Ingrid used coefficients of the Mallat transformation to capture the directions and spatial distributions [22]. The principal component analysis (PCA) model has simple construction via singular value decomposition (SVD), providing the best low-dimensional linear approximation of the data, and the time lag shift [23–25].

Data reduction

The data reduction technique can be used to obtain a reduced representation of the dataset, which is much smaller but still close to preserving the integrity of the original data and producing the same or nearly the same results as the pre-reduction ones. The effective range of data reduction is well-known as reducing the storage capacity for a given dataset, and this includes eliminating invalid instances in a data archive and producing the summarized statistical aggregation of data attributes in the columns of data. Technologies like compression (decreasing the space required to store a block of data), deduplication (eliminating the redundant data from multiple data sources), thin provisioning (a dynamic strategy for allocating storage address while removing the previous location of unused data) and efficient snapshots (using some synopsis data structure containing time copies of data and changes of blocks) are employed in the strategies to deal with reduction problems. Andrews, H., and C. Patterson gave the combination of SVD and clustering for reducing the data dimensions [26]. Shearer Colin proposed CRISP-DM (cross industry standard process for data mining) project to distinction of generic and special process model structures for reducing the high dimensionality [27]. Glas, Annuska M., et al. demonstrated the reproducibility and robustness of the small custom-made microarray as a reliable diagnostic tool [28]. Yoshida, Hisako, et al. created the algorithm according on radial basis function-sparse partial least squares (RBF-sPLS) by creating sparse, linear combinations of explanatory variables so that can concurrently perform feature selection and dimensionality reduction [29].

Classification

In data mining, classification is one of the most popular tools for understanding relationship among various conditions and the features of different objects. Classification methods can help identify a new observation belonging to which particular part of categories (sub-populations), on the basis of a training set of data containing observations (or instances) whose category membership is known in advance. In this section, we demonstrate some typical classification techniques, such as k-nearest neighbor (KNN), Naïve Bayes (NB), decision tree (DT), support vector machine (SVM), neural network (NN), and ensemble (EM).

K-nearest neighbor

As a type of lazy learning methods based on instances, the unclassified data point is discovered and assigned to particular label according to the formerly known k nearest neighbor (KNN) points and a voting mechanism is utilized during the determination of the target object belonging. KNN together with linear discriminate analysis (LDA) and sequential forward selection (SFS) were used as main body of classification by Jen, et al. to study the relationships among critical factors between both health people and those with chronic illnesses based on characteristic value determination. Early warning system was then followed to classify and recognize the chronic class [30]. Weighted KNN, proposed by Bailey and Jain [31], was improved with strengths of non-uniform distribution through assigned weights to k neighbors when each distance of data point calculated. Keller, et al. developed a fuzzy version of KNN to give unequal importance to the predicted samples in determining the classified memberships of certain typical patterns [32]. An enhanced fuzzy KNN model for classifying thyroid disease was presented by Liu, et al. [33]. It used one kind of metaheuristics, namely particle swarm optimization (PSO), to specify the neighborhood size k and the fuzzy strength parameter m. Principle component analysis (PCA) was also involved in the effectiveness validation of discriminative subspace building during the classification. The output showed an adaptive manner of such kind of enhanced fuzzy KNN. Sometimes similarities among data or overlap of majority classes may cause misclassification when voting, so Syaliman, et al. [34] proposed local mean based and distance weight(LMDW) KNN to make higher performance during the majority vote phase. The two efficient and methods were merged together to output improved results than usual.

Naïve bayes

The Bayes' theorem is the foundation of the naïve Bayesian (NB) as a probabilistic statistical classifier, whereas the so called naïve or simple assumption is made that attributes or features are conditionally independent, so that computational complexity can be decreased during the multiple operations of probabilities. Spiegelhalter, David J., et al. applied Bayesian probabilistic and statistical ideas to expert systems that exacts probabilistic inference on individual cases, possibly using a general propagation procedure [35]. Kononenko, Igor optimized the tradeoff between the "non-naivety" and the reliability of approximations of probabilities, and presented a set of diagnostics for identifying conflicts between data and prior specification by using Bayesian statistical techniques [36]. Langley, Pat reviewed the induction of simple Bayesian classifiers that involves disjunctive concepts, since they violate the independence assumption on which the latter relies [37]. Peng, Hanchuan, and Fuhui Long designated a heuristic algorithm, Bayesian metric, to discrete variable with a limited number of states and learned the ψ-structure of belief network (or Bayesian network, BN), where clusters of "equivalent" pixels are regarded as the irregular features [38]. More recently in Hickey's research, the combination of greedy feature selection and Bayes classifier was used on a robust public health dataset to test one or some features which can forecast the best target. The innovation part in their work was reflected in the imprecise search of attribute space without seeking the whole exhaustively, and also the random weighted selections with ranking scores of plentiful attribute probabilities [39]. Except for the greed feature ranking method in NB classifiers, a new quick variable selection way was designed by Joaquín Abellán and Javier G. Castellano [40] to overcome the disadvantages of non-negative and preset information threshold in Info-Gain computing in a NB model. The most informative variables were chosen by imprecise probabilities and the maximum entropy measure without the setting threshold. The result was proved to be a valuable tool for processing bioinformatics benchmark datasets, especially there are volumes of complex features and tremendous instances in it.

Decision tree

Classification learning of observations in decision tree (DT) goes from features (denoted by nodes) to outcomes of targeted class values (represented by leaves) through logical conjunctions of those attributes (calculated by branches), as a whole flowchart of decision support. The crucial strategy behind a decision tree is the traditional top-down divide-and-conquer approach utilizing information entropy of different features.

Estella, Francisco, et al. designed and implemented a system in order to manipulate magnetic resonance images (MRI), store big data for patients, and then used fuzzy decision tree (FDT) classifiers to achieve feature extraction as well as feature selection combination for decision-making in classification task [41]. Rodriguez, Juan José, et al. proposed a decision trees generated classifier ensembles based on feature extraction in order to preserve the variability information in data. The feature set randomly splits into K subsets rotations, then takes place to form the new features for a base classifier and PCA is applied to each subset [42]. Domingos, Pedro, and Geoff Hulten proposed an efficient decision trees based on the ultra-fast decision tree, called concept-adapting very fast decision tree CVFDT. The idea is sliding window of current examples that every time a new example arrives with O(1) complexity per example, as opposed to O(w), where w is the window size [43, 44]. In Zhu, et al.'s study [45], when determining the splits in tree branches, the optimal split scoring function estimation was taken to replace maximum likelihood estimator (MLE) in the related calculation in entropy. The fact they found was that sub-sample number decreased rapidly with the trend of exponential form during node splitting, thus the assumption of single MLE (sample size is much more than support size) was no more eligible. Their method was validated on thirteen different entropy estimation schemes over a set of benchmark datasets including bioinformatics. Taking account of the sensitive costs of acquired features and misclassified instances, Esmeir and Markovitch used anytime cost-sensitive tree (ACT) to generate a sampling collection of sub-trees for every feature, the stochastic tree induction algorithm was employed to compute the minimum cost as an estimated value for attributes [46]. Several years later, they persisted in the work of stochastic approach of improving ACT with a novel framework for tree classification atanycost (TATA) [47], additional resources can be exploited to generate a classifier at any cost during learning time, when the time itself would be pre-allocated or determined dynamically. The budget of classification tasks was also considered as well, so overall it was resource-bounded.

Support vector machine

The pivotal mind of support vector machine (SVM) is the hyper-plane constructed by implicitly mapping the original input space to higher dimensional one in order to make the distance between two separated classes as maximal as possible. For the mapping step, it is usually done via user specified kernel functions like radial basis function (RBF), sigmoid, Gaussian, polynomial, etc. Bounds on the generalization performance based on the leave-one-out method and the VC-dimension are given to match the complexity of the problem and automatically adjusted the effective number of parameters [48]. Lee, Ki-Joong, Young-Sook Hwang, and Hae-Chang Rim took a two-phase named entity recognition method based on SVM (one is to pinpoint entities by a simple dictionary look-up, and another is to classify the semantic class of the identified entity by SVMs) and dictionary that can alleviate the entity identification and the semantic classification, the unbalanced class distribution problems [49]. Nanni, Loris, et al. used a local binary pattern (LBP) approach based on uniform local quinary pattern (LQP) and a rotation invariant local quinary pattern (where a variance bin selection is performed and neighborhood preserving embedding feature transform is applied) to find the best way for describing a given texture [50]. Hasri, et al. proposed multiple SVM - recursive feature elimination (MSVM-RFE) as a gene selection to identify the small number of informative genes over leukemia and lung datasets. Their idea was straightforward for reducing the dimensionality of a subset of original data and repeating the selection operation on a few subsamples from bootstrap re-sampling on original data to stabilize the performance of SVMs [51]. In the research work of Kavitha, et al. [52], a boosted SVM version named fast correlation-based filter (FCBF) SVM was introduced based on MSVM-RFE, the authors thought that FCBF should be added first to select the most prominent not correlated genes and reduce further dimensionality than MSVM-RFE. More accurate output and less computational time were acquired as an inspiring result.

Neural network

Inspired by the physical neural network in biological field, neural network (NN) is designed to simulate neurological function system that has multiple layers of grouped and interconnected processing nodes known as neurons with adjustable weighted links, aiming at addressing the particular classification issue under the entire unity of aggregated neurons. The training and learning process is reflected in adjustment of linkage weights and changes in network structure as an adaptive manner in terms of the information flowing internally and externally via network during studying steps. NN is also referred to artificial neural network (ANN) frequently. The most commonly used NN model is a typical multilayer perceptron (MLP) with backpropagation (BP), which feeds forward the values first and then calculates gradient of the loss function of each neuron error contribution, and at last propagates it back to the earlier layers. The iterative gradient descent optimization can lead to reasonable classification results of new types of data, but cause large expensive costs of parameter choice and model training. Chest diseases consisting of asthma, pneumonia, tuberculosis and other chronic pulmonary diseases are analyzed in a comparative study using four types of NNs, they are multilayer (MLP or MLNN), generalized regression (GRNN), probabilistic (PNN) and learning vector quantization (LVQ) [53]. The respective structures and properties of those NN models were discussed by Er, et al. in a detailed way and experimental results reported that NN could definitely help diagnosis of chest diseases as components of learning based decision support system and each model has its own power for dealing with specific disease dataset. Gunasundari, et al. believed that NNs are method-free tools capable for identifying disease types, i.e. no special given algorithm is needed to particular disease diagnosis [54]. In their research work, lung cancer tissue was discovered and segmented from chest computed tomography (CT) as a feature extraction phase to minimize analyzed data size and NN was applied to tell the various lung diseases apart. Bin Wei and Jing Zhao proposed a metaheuristic based novel NN architecture for precisely detecting exon intron boundaries and splice sites prediction in gene identification [55], the improved particle swarm optimization (IPSO) gave their method the ability of accelerating the convergence and avoiding local optimum during training NN structure and building the model.

Ensemble

Ensemble (EM) can be considered as a hybrid of various approaches with the reasonable and logical hypothesis behind that many classification models are able to work collaboratively and get superior classification outcomes than single one only. As a typical case of ensemble learning, random forest (RF) is an ensemble classifier made up of various decision trees to be a forest with outputs of major classes of all individual trees. As an expansion of improved RF, enriched random forest (ERF) was proposed by Amaratunga, et al. [56] to use the weighted random sampling for selecting the desired subsets at each tree node and the weights were tilted according to informative features. The test results showed a superior performance on microarray datasets. Similarly, Yao, et al. presented another enhanced RF using a replaced sampling method and multiple example subsets were randomly extracted with replacement from major class to be balanced with the minor class [57]. Fabris, et al. studied the rule based method "computing the predictive accuracy of random tree rules with positive (±) feature values (COMPACT + FV)"to measure the importance of each positive feature value in a RF model. They believed that considering fewer feature values is better than measuring as a whole set [58]. Gopal, Ram, et al.'s special issue contribution, "Multi-Objective Design of Hierarchical Consensus Functions for Clustering Ensembles via Genetic Programming," focused on adding the "Technique" and "Task". They presented a novel genetic programming (GP) based approach formed as a hybrid of advanced strategies of multi-objective clustering and clustering ensembles [59]. Ding, Jing, et al. proposed a classifier ensemble that can alleviate these problems by compensating for the weakness of a classifier with the strengths of other classifiers, assuming that the errors made by individual classifiers are not fully correlated [60]. Shen, Hong-Bin, and Kuo-Chen Chou introduced ensemble classifier that is composed of a set of basic classifiers (optimized evidence-theoretic k-nearest neighbors), with each trained in different parameter systems, and the outcomes are combined through a weighted voting to give a final determination for classifying a query protein [61]. Eom, Jae-Hong, et al. combined a set of four different classifiers (SVM, NN, DT and BN) with ensembles for supporting the diagnosis of cardiovascular disease (CVD) based on aptamer chips [62].

Clustering

Clustering is the task that partitions sample data into clusters, and groups a set of objects in such a way that members in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups [63]. In this section, we demonstrate some typical clustering techniques, such as hierarchical clustering (HC), partitioning relocation clustering (PRC), density-based clustering (DBC), grid-based clustering (GBC), and model-based clustering (MBC).

Hierarchical clustering

Hierarchical clustering (HC) gathers data objects into tiny clusters, where those smaller clusters are categorized into larger clusters from button to up layer, and so forth in the hierarchical manner. Zhang et al. proposed a distance measurement that enables clustering data with both continuous and categorical attributes, namely BIRCH that is especially suitable for very large datasets [64]. Bryant, David, and Vincent Moulton illustrated a hierarchical distance-based approach named Neighbor-Net, which provides a snapshot of the data for constructing phylogenetic networks and guiding more detailed analysis. The Neighbor-Net is based on the Neighbor-Joining (NJ) algorithm [65]. Heo, Moonseong, and Andrew C. Leon derived a mixed-effects linear regression model of three hierarchical-based cluster randomized clinical trials, which can determinate the randomly assigned sample size and require to detect an intervention effect on outcomes at health care subject's level [66]. An accelerating version of hierarchical clustering of microarray gene expression data was carried out by Darkins, et al. [67]. The Bayesian hierarchical clustering (BHC) strategy was applied as a randomized statistical method using Gaussian process to build more flexible models and handle wide ranges of unstructured data more adaptively. With the rapid development of bio-inspired optimization methodologies, the communicating ants for clustering with backtracking strategy (CACB) algorithm first came into being in Elkamel, et al. [68]. It can ensure multi clusters aggregation simultaneously and result in a compact dendrogram, its natural properties were dynamic and adaptive thresholding policy and backtracking strategy in former aggregation steps. However, Pelin Yildirim and Derya

Birantargued that randomization may affect and discriminate the clustering results seriously [69],so by introducing two new concepts k-min linkage (the average of k closest pairs) and k-max linkage (the average of k farthest pairs), they proposed k-linkage agglomerative hierarchical clustering to measure the distance of k observations from two clusters separately. The method showed a deterministic result and higher accuracy than traditional HC.

Partitioning relocation clustering

Partitioning relocation clustering (PRC) organizes and constructs several segmentations of data, and each partition is a subgroup of original collection of data and stands for every individual cluster. The iterative relocation technique is the rationale key idea of it, attempting to modify suitable number of partitions finally. Chiu, Tom, et al. proposed a clustering algorithm using the distance measurement based on the framework of BIRCH that performs a pre-clustering step by scanning the entire dataset and storing the dense regions of data records in terms of summary statistics, and enables the algorithm to automatically determine the appropriate number of clusters and a new strategy of assigning cluster membership to noisy data [70]. Hussain, Hanaa M., et al. proposed a parallel process approach, which is Xilinx Virtex4 XC4VLX25 field programmable gate arrays (FPGA) to accelerate the five K-means clustering cores for processing Microarray data [71]. Tseng, George C. gave a K-means clustering approach, which extended on penalization and weighting to avoid scattered objects clustering, account and identify for prior information of preferred or prohibited cluster patterns in meanwhile [72]. Botía, et al. brought extra K-means processing step into weighted gene co-expression network analysis (WGCNA) and created k-means to gene co-expression network (GCN) for assessment on UKBEC and GTExhuman brain tissue data. It refined the output of WGCNA with K-means as a hybrid post-processing [73]. Due to the truth that ultimate cluster distribution is heavily relied on the random positions of initial centroids and insufficient improvements of additional input information about data points, Sathiya and Kavitha invented an enhanced initial cluster centers based K-means approach with the core technique to reserve some informative knowledge using simple data structure in current and next iterations [74]. The popular variants of K-means such like K-medians (calculating median as the centroid in place of mean), K-medoids (removing outliers towards sensitivity in K-means) and K-modes (matching dissimilarity measure to replace Euclidean distance metric and using mode as the centroid) are reviewed in [75].

Density-based clustering

Instead of measuring distance as the criteria of cluster partitioning, the density of data point distribution within the given radius of neighbors is used for arbitrary shaped clustering in density-based clustering (DBC). Jiang, Daxin, Jian Pei, and Aidong Zhang used a density-based, hierarchical clustering (DHC) approach to identify the clusters, which tackled the problem of effectively clustering and improved the clustering quality and robustness [76]. Kailing, Karin, Hans-Peter Kriegel, and Peer Kröger introduced an effective and efficient approach, SUBCLU (density-connected subspace clustering), which underlay the algorithm DBSCAN to detect arbitrarily shaped and positioned clusters in subspaces and efficiently prune subspaces in the process of generating all clusters in a bottom up way [77]. Another issue of human intervention in the workflow of DBSCAN when choosing the input parameters was resolved by using the bio-inspired cuckoo search algorithm in Wang, et al.'s publication [78]. The global parameter Eps was optimized and the clustering procedure became automatic. Günnemann, Stephan, Brigitte Boden, and Thomas Seidl introduced a density-based cluster, DB-CSC, to detect clusters of arbitrary shape and size, and avoid redundancy in the result by selecting only the most interesting non-redundant clusters. This approach not only circumvents the problem of full-space clustering, but also limitless cluster of certain shapes [79]. A robust density-based clustering (RDBC, by Florian Sittel and Gerhard Stock [80]) with properties of deterministic, efficient, and self-consistent parameter selection was presented to determine protein metastable conformational states, in terms of space density under a geometric coordinate system. And then the definitions of local free energies were set up and Markov state models were built from molecular dynamics trajectories to generate superior effects on the data. Similarly, Liu, et al. [81] implemented the adaptive partitioning of local density-peaks of DBC, the application as also for molecular dynamics trajectories analytics. The simple KNN search was performed and molecular dynamics conformations densities and locations were obtained for next step of grouping into clusters. Such density-dependent mechanism can guarantee the cluster size to be adaptively scalable in different regions.

Grid-based clustering

Multi resolution grid structures of the space of objects are formed into independently quantized cells, wherein all calculations of clustering steps are implemented in grid-based clustering (GBC). Maltsev, Natalia, et al. illustrated a grid-based high-throughput PUMA2 interactive system, which allows users to submit batch sequence data for automated functional analysis and construction of metabolic models to compare and evolve analysis of genomic data and metabolic networks in the context of taxonomic and phenotypic information [82]. Ortuso, Francesco, Thierry Langer, and Stefano Alcaro provided a GRID-based pharmacophore model (GBPM), which can define most relevant interaction areas in complexes deriving GBPM from 3D molecular structure information. It is based on logical and clustering operations with 3D maps computing, which can share, standardize and give a reliable solution for biological data storage and analysis [83]. Porro, Ivan, et al. implemented Grid based platform, Grid portal, which contains three analysis sequence (group opening and image set uploading, normalization, and model based gene expression) to hide the complexity of framework from end users and to make them able to easily access available services and data [84]. Ren, et al. [85] proposed PKS-tree based GBC that can tackle massive dimensional stream data clustering efficiently in the aspects of storage and indexing. Non empty grids were recorded by PKS-tree as well as the empty relationships between those grids, the new arrived data point was mapped and inserted into a grid according to the density relationship stored in PKS-tree. Such algorithm can reduce the time complexity to O(log N) at most. And for Liu, et al. in [86], they gave out a hybrid approach combining density with grid-based clustering, where grid parameters were used to determine the data space division into the valid or invalid meshes, then valid meshes got merged in the direction from up-left to bottom-right of the whole diagonal grids of the located data points. Meanwhile for invalid grids, they were searched by the boundary grid values. Finally only two adjacent grids can be processed at a time, and all the valid ones located in diagonal-direction grids. The results demonstrated the reduced clustering time and improved accuracy than traditional methods.

Model-based clustering

The basic idea for model-based clustering is the supposed model for each cluster firstly and the replacement of best fitting model for the given one. Two technical foundations of MBC algorithms are employed: statistical learning method and neural network learning approach. Expectation-maximization (EM) analysis based on statistical modeling iteratively refines the weighted representation of belonged memberships and assigns an object to it according to cluster mean values. Si, et al. applied EM for clustering model parameters estimation on RNA sequential data [87]. Moreover, two versions of stochastic EM algorithms were also introduced to jump out of the local trap and intended to find the global solution. COBWEB is built on heuristic inspired standards for classification tree (differs from decision tree with probabilistic description of concept) formation of hierarchical clustering. Abawajy, et al. evaluated the outcomes of multi-stage means for ECG data clustering partially, COBWEB, EM, Farthest First and K-means algorithms were systematically compared to get the conclusion that all those methods can independently do data processing without extra information but depend on initialization of selecting parameters because of stochastic nature [88]. Self-organizing map (SOM) with NN architecture of an exemplar for each cluster prototype, constructs feature mappings of high dimensional input space to low dimensional output space, the reduction is done but at the same time distance relationships of the topology are kept possibly as well. Wang, et al. conducted series of DNA microarray experiments to uncover the hidden patterns of gene expression data by mainly usage of SOM [89]. Simplified component plane was revealed via SOM and the further clustering of SOM results demonstrated the feature patterns.

Data mining has a relatively wide range of applications in bioinformatics since it offers many practical benefits by improving processing accuracy, saving precious diagnosis time, assisting to make valuable decisions, maximizing revenues, etc. Following are those mentioned above typical data mining techniques applied in bioinformatics via many ways, as shown below (Table 1):

In the next table (Table 2), we list the advantages and disadvantages of data mining techniques. Different learning approaches may focus on handling a certain biomedical area. The comparison can give a guide to select appropriate methods for processing the bioinformatics.

Deep learning in bioinformatics

Over the past decades, data mining approaches were primarily applied on traditional knowledge discovery in well-structured relational database with mainly numerical data, in order to meet the demands of industrial business requirements. The applied data analysis methods were dominated techniques like statistics, simple logical reasoning, etc. Current situation for data mining field lies in the exploration of heterogeneous data besides homogeneous one (involving the forms of structured, semi structured and unstructured) by utilizing core thoughts and processing tools of machine learning, pattern recognition, artificial intelligence, etc. to perform more intelligent learning strategies and obtain potentially undiscovered knowledge just like our human beings. As for the future trend of data mining domain, in addition to data structure issue, much more complicated data objects will be dealt like big data with homogeneous types, large noises, multiple representations, huge volumes and high speed streams. Apart from the consideration of big, deep is worth taking into account as well through sophisticated technologies of deep neural network (DNN), parallel and distributed computing, metaheuristics, fuzzy logic, cloud computing, etc. It is quite obvious that data mining tendency not only serves as the powerful processing mechanism of big data, but also provides more deep insights of knowledge extraction in kinds of scientific and research fields.

Table 1 Review of various data mining techniques in biomedical analysis

Technique	Application aim	Source data
KNN	Hypertension warning [30]	Cardiovascular disease data
NB	Prognostics of breast cancer recurrence [36]	Breast cancer data
DT	Decision-making in cancer classification [41]	Magnetic resonance images
SVM	Identify entity into its semantic class [49]	Protein, DNA and RNA's subclasses
NN	Algorithm-free disease identification [54]	Chest CT
EM	Diagnosis cardiovascular disease [62]	Aptamer chips data
HC	Gene conversion [66]	Archealchaperonin sequences
PRC	Investigate high-throughput biological data [72]	Genomic and proteomic data
DBC	Investigate high-throughput biological data [76]	Time series gene expression data
GBC	Bioinformatics matching [82]	Genomic data
MBC	Automatic clinical diagnosis [88]	ECG

It has been extremely accumulated and developed for biomedical data in an unprecedented way of wide extent and depth nowadays. And it leads to large amount of machine learning algorithms for mining available knowledge from

Table 2 Comparison of advantages and disadvantages of various data mining approaches

Technique	Advantages	Disadvantages
KNN	Simplicity and fast speed of accomplishment	Noise sensitively aware
		High space complexity required
NB	Less time consumption and higher accuracy for huge dataset	Cannot give accurate results if there exists dependency among variables
DT	No domain limitation of the knowledge to construct decision tree	Restrict to one output attribute
	Can easily process the data with high dimension	Performance of classifier is depend upon the type of dataset
	Can handle both numerical and categorical data	
SVM	Better accuracy compared with other classifiers	Should select different kernel function for different dataset
	Easily handle complex nonlinear data points	The comparison with other methods training process takes more time
NN	Can simulate almost any functions for complex applications and problems	The black box nature, hard to interpret the structure
	High performance of accuracy	Large computational cost
	The availability of multiple training	May over-fitting after times of training
EM	Improvement in predictive accuracy	Difficult for understanding
		Using the wrong ensemble method will get bad results
HC	Flexibility regarding the level of granularity	Difficult for choosing the criteria bound
	Easily to handle the type of similarity or distance problems	Cannot re-construct clusters
	Applicability to any attribute type	Quite heuristic and informal
PRC	No limitations on attribute types	Not obvious to choose K value
	The choice of medoids is calculated by the location of an inside cluster	Only numerical attributes are covered
		Resulting clusters can be unbalanced
DBC	No need to set the cluster number in advance	Fail in varying density clusters
	Able to identify noise data while clustering	Not work well in high dimensional data
	Able to discern arbitrarily clusters shape	
GBC	Able to work well in relatively high-dimensional spatial data	Strong dependency on cluster initialization
	O(N) linear time complexity	Dependency on parameters setting
MBC	Formal models with explicit statistical properties and standard distributions	Sometimes models seem to be too simple for real world application scenarios
	Adaptive in selecting model parameters flexibly	May cost a lot of computing resources especially for NN structure

bioinformatics. Meanwhile, broad boundaries of practical applications range from many real world scenarios, among which massive scale of data also brings challenges including stronger learning adaptability for huge volume and high dimensions, superior process ability for heterogeneous nature of data, etc. More recently, deep learning evolves from traditional neural networks (NNs) in the branch of machine learning techniques with the basic concept of neuron processors and the essential architecture of multiple processing layers for the sake of transferring non-linear relationships through responses of each layer [90]. It facilitates primary and successful applications in majority of areas like speech and image recognition, natural language processing, biomedical research with strengths in up level abstractions of features of large raw data, distributed and parallel computing, sophisticated learning mechanism without too many manual interventions instead.

Deep neural network

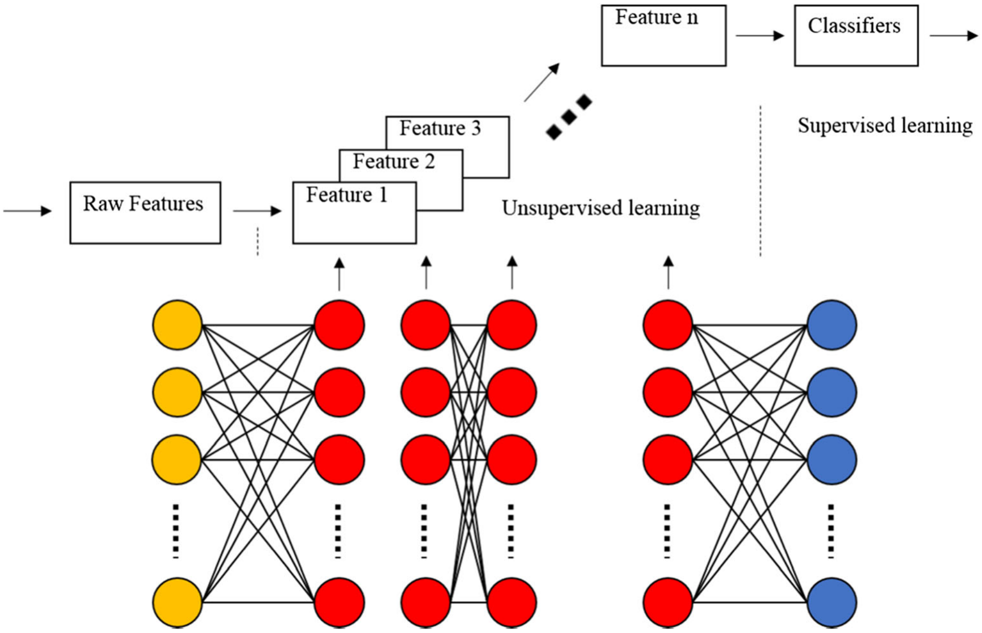
At first sight, the obvious characteristic of a DNN architecture is hierarchy forms which mainly consists of layers of input, multi hidden and output (Fig. 1). The output result is computed straight forward along the sequent layers of DNN as long as input data is fed, and this kind of working mechanism in a neural network is known as feed forward. In each neuron of middle hidden layer, the output result in vector format from previous layer is multiplied by a weight vector and plus a bias value in the current layer, then the biased weighted sum is put into a nonlinear function (sigmoid, hyperbolic tangent, or rectified linear unit (ReLU), etc.) to get a number as the output of this cell at present. Tremendous neuron outputs in the same hidden layer comprise a new numeric vector called feature layer. From a macro scale view the primitive patterns with raw features are transformed to more abstract feature representation levels through such kind of deep processing steps to reach the final goal of classification task. It is such hierarchical representation learning method that makes it successfully find out desired but abstract underlying correlations and patterns among large amount of bioinformatics data and offer a meaningful insight of understanding bioinformatics better.

According to the types of layers and the corresponding learning methods, there are many deep learning networks of DNNs, among which typical examples are deep belief network (DBN) [91] and stacked auto-encoder (SAE) [92], as well as Convolutional neural network (CNN) [93] and recurrent neural network (RNN) [94]. Those models are the most widely used in biomedical analysis with a certain representative property of model structure and training process.

Convolutional neural network

CNN is commonly used in image processing, recognition, computer vision and other areas, especially two-dimensional signal processing, its idea stems from the study of optic nerve, in which the discovery of principle of receptive field [95] is greatly significant to the CNN model. As a significant member of DNNs, there are different hierarchical layers that comprise the main body structure of CNN model. Inspired by the study of visual cortex of the brain, two further groups of cells are classified as simple neurons and complex neurons, where simple ones receive the raw signals within a particular area of sub-regions in visual stimuli and the complex group copes with the subsequent synthesized information from simple group to meet the requirements of more intricate process.

Fig. 1 Demonstration for general structure and construction process of DNN in each step



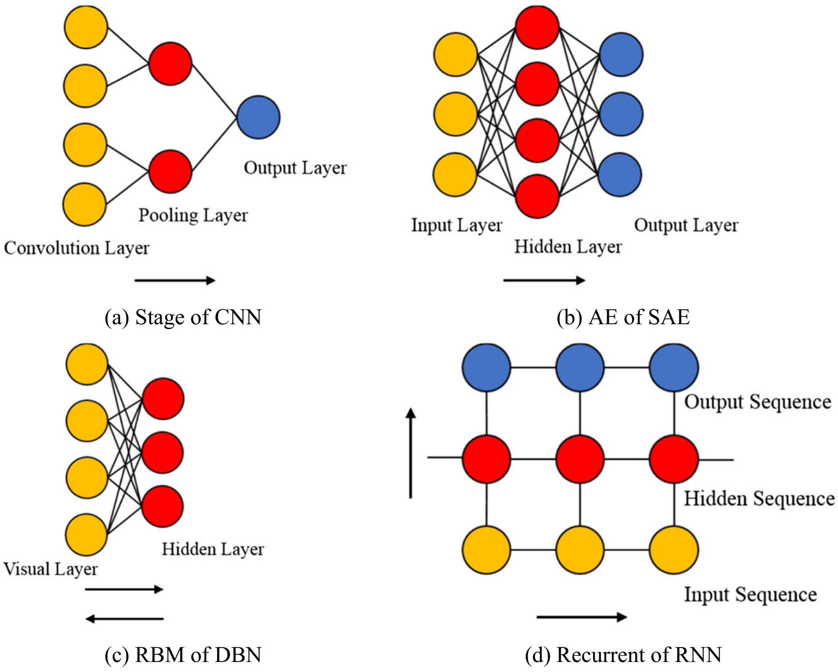
Thus, CNN mimics the function flow of the brain and acts as a prominent deep model with those key ideas: multi-layer stack, regional connectivity, weight sharing and pooling.

This deep CNN model consists of multiple "stage" stack illustrated in Fig. 2a, each basic component "stage" contains a convolution layer and a polling layer. The convolution layer can capture feature maps form regional connectivity by doing convolution between small regions and using the weighted mask in that area, thus hyper parameters can be essentially reduced with application of the principle of weight sharing. The pooling layer merges the adjacent nodes into one to acquire similar but more aggregated and complex features, moreover reducing the amount of training data. After constructing several stacked stages together, a number of fully connected layers and classifiers are added at the end to further process data, which make it possible for non-linear abstract representations. Finally, the whole CNN model is trained via supervised learning.

Due to the eminent reputation of excellent performance of successful deep architectures for analysis on bioinformatics data, CNN has contributed immensely in the primary research fields and corresponding domains. The straightforward applications of CNN can be fulfilled since it has the potentiality not only in a one-dimensional signal interval (for example genomic sequence) to retrieve the recurring patterns, but also in a two-dimensional image grid (such as mammography) to extract the learned features, or even in a three-dimensional data cube (for instance 3D MRI) to obtain the spatial outlines. The most distinguishing advantage of CNN is automatic feature representation of a tensor form for the input dataset no matter what the task is, and furthermore there are only few parameters in the CNN model compared to a fully connected NN because of the local connectivity and weight sharing with the same amount of neurons in hidden layers. But it requires a lot of memories and resources when computing the intermediate values of hidden layers and constructing the fully connected classifier in the last step.

Stacked auto-encoder

SAE's basic building block is called auto-encoder (AE), which has the gradual stacked structure of three layers: input, hidden and output. The core idea behind auto-encoder relies on the concept of a well-constructed model representation with the built encoder and decoder. The encoder has almost the same function in DNN as transforming the input vector to a hidden layer representation with weight matrix and bias/offset vector. Simultaneously the decoder maps the hidden layer representation to the reconstructed input, which is regarded as the output result. The topological diagram is shown in Fig. 2b. AE model sets the training target to fit the input data, (i.e. the network output is as much as possible equal to the input), and then is trained by backpropagation algorithm, by feeding input and setting the difference error between reconstructed input and original one. Although the training process of AE is supervised, it does not require the original data to be classified and labeled, so the whole training process is still unsupervised. If sparse penalty (the number of activated cells in the network is limited or constrained) is introduced to the training, then a sparse AE is formed [96, 97]. Suppose random noise is combined with the input data during training, the de-noising AE is obtained [98], whereas



(a) Stage of CNN

(b) AE of SAE

(c) RBM of DBN

(d) Recurrent of RNN

Fig. 2 Illustrations of architecture and topology of each basic unit, making up the four different types of DNNs: **a** stage of CNN; **b** auto-encoder of SAE; **c** restricted Boltzmann machine of DBN; **d** recurrent unit of RNN

these two models can often learn better characteristics of data in practice.

The training process of SAE is similar to that of CNN, and the procedure is demonstrated in (Fig. 1). After training the first AE unit, the hidden layer is set to be input of next layer AE, which is trained in same way as the previous one. Through this iterative way, the final layer of SAE is the abstract feature of the input data after several transformations. At last, according to situations of the problem, different output layers are connected, and the weights of output layer are trained by supervised learning algorithm, and then final classification result is performed. The benefits that SAEs provide are listed out below: its layer wise working strategy becomes very much suitable as far as neural network structure is concerned, and no matter whatever the input data type is, SAE would simply replace the loss function with a new one and change the relevant activation function. It needs not to reconstruct the total network using a lot of time and resources, but may only fine-tune via the back propagation algorithm.

Deep belief network

DBN is composed of multiple restricted Boltzmann machines (RBMs) as each single RBM contains a visual layer and a hidden layer, a two way direction link between two layers, in which the visual layer simultaneously works as input and output multiplexing displayed in Fig. 2c. Having the characteristics of highly complex directed acyclic graph in the form of sequential RBMs, DBN is trained through building up RBM layers level by level instead. RBM model refers to contrastive divergence algorithm to train unlabeled samples without high cost of training expense, so it belongs to unsupervised learning algorithm. The final model can restore the visual layer data from the hidden layer data due to bidirectional property, thereby the hidden layer is the abstract expression of the visual layer. DBN construction process is like SAE as follows: First training to get the first RBM, then freezing weights and setting the hidden layer as the next RBM's visual layer, with the same training method to get the second RBM. Recursively, a number of stacking multiple RBMs together in sequence forms a deep Boltzmann machine (DBM). In the top of the DBM, a layer namely associative memory is settled to convert DBM to DBN. Resembling SAE, the output will be multi-level abstraction of input data after layer-wise unsupervised learning of patterns. Good results are usually got when applying classifiers to highly abstract features. Recently, a hybrid method adding convolution layer before DBN to form a new convolution DBN receives successful application results in face recognition [99] and audio classification [100].

Since DBN training produce basically consists of rapid RBM modeling of contrastive divergence algorithm, it illustrates the capacity to quickly derive an optimized parameter set from a large number of search spaces. DBN also shows the ability for addressing low speed of convergence and local optimum problems and calculating the outcome results of the variables in each layer in a manner of approximately inferring way. However the drawbacks of DBN exist in the limitations of such approximate inference method, because it is a quite greedy algorithm working within every layer once a time and there is a lack of reactions between other layers for parameter optimization.

Recurrent neural network

RNN with a fundamental structure of recurrent and cyclic connection unit is designed for handling sequential information and therefore it has the ability of memory (Fig. 2d). Recurrent computation is conducted in hidden layer while processed input data arriving sequentially, causing old data remembered implicitly in state vectors (units in hidden layer). So it is a further extension of traditional feed forward NN with the fact that hidden unit relies on the computation of current input and previous values stored in those state vectors [101], output is affected by both past and current inputs consequently, whereas in a conventional NN the relationship of inputs and hidden neurons is independent for each other. At a first glance, RNN seems to be not as deep as those three multilayer models mentioned above, but its memorized storage nature leads to even deeper architecture if a long period of time is considered. One serious problem for RNN is vanishing gradient [102], losing necessary information rapidly based on transmission of values in activation functions because of the simple one nonlinear layer in each RNN cell. Fortunately, long short term memory (LSTM) [103] or gated recurrent unit (GRU) [104] can help to prevent using substitution of simple hidden units with more complicated ones. Compared to RNN, those improved networks have the complex structure by adding more components in their recurrent unit such as input/forget/output gates, cell/hidden transmission states, etc.

Although RNN acquire less attentions and explorations than CNN, it has been proved that RNN still has the dominated power to process and analyze sequential data or since it can usually map variable-length input sequence to another fixed-size sequence and make the consequent predictions via its recursive loops. RNN is not the first primary option for biomedical imaging processing, but it is also believed that RNN can perform a good incorporation with CNN and enhance its influence in the corresponding research domain.

After describing different sorts of frameworks and architectures in a theoretical way, there are two major fields for deep learning applied in bioinformatics: medical (disease diagnosis, clinical decision making, medical image processing, drug discovery and repurposing) and biological (genomics, transcriptomics, proteomics) data analysis. The following (Table 3) lists out a detailed review of success of deep learning applications in reality.

Table 3 Review of various deep learning techniques in biomedical analysis

Technique	Application aim	Source data
SAE + softmax	Classification and detection of cancer [105]	Gene expression data
DBN	Clustering of cancer [106]	Breast and ovarian cancer
CNN + SVM	Cataracts grade [107]	Cataracts data
CNN	Medical image auto segmentation [108]	CT + MRI
RNN	Prediction of protein contact map [109]	ASTRAL database
RNN + DBN	Classification of metagenomics [110]	Microbiome sequence data
SAE	Medical image reconstruction [111]	MRI
RNN	Prediction of liver injury drug induced [112]	Drug data

Here, an initial evaluation of popular deep learning frameworks is shown (Table 4), making it feasible and easy implemented for researchers or workers in the industry. All the frameworks are compute unified device architecture (CUDA, GPU programming) acceleration supported.

Data mining and deep learning comparison

Data mining, as its name suggests, is to dig hidden information from massive data. According to its definition, the mined object here is a large number of incomplete, noisy, fuzzy, and random practical application data. The information refers to implicit, regular, previously unknown, but potentially useful, and ultimately understandable knowledge. In a real business environment, data mining is more application-oriented. As a common method to fulfill data mining technologies, machine learning is the use of computers, probability theory, statistics and other knowledge. By inputting data into computer programs and letting computers learn new knowledge, it is a way to realize artificial intelligence, but this kind of learning will not let the machine generate consciousness. The process of machine learning is to find the objective function through training data. Data quality can affect machine learning accuracy, so data preprocessing is very important. Deep learning, which is a new type and field of machine learning, is motivated mainly by establishing, simulating the thinking of the human brain, and analyzing the neural network of learning. It can be said to be a kind of brain, which mainly imitates the mechanism of the human brain to interpret data, such as image, voice, text and signal.

Till now to sum up, we list out the key ideas and basic components behind various data mining and deep learning algorithms in Table 5 for readers to understand them intuitively. From technical point of view, machine learning is the core technique of data mining, and now it is very natural to make the comparison of machine learning and deep learning from some aspects discussed below:

- **Material Dependency**: Among many dependent factors for implementing algorithms, the first significant matter that traditional machine learning differs from deep learning is the scale of data volume. Traditional data mining mainly focuses on a relatively small amount of high-quality sample data. Machine learning is more about massive and mixed data. However, machine learning does not necessarily require global data. It is only in the era of big data that the methods of huge

Table 4 Comparison of most popular frameworks for deep learning

Framework	Language	Advantages	Model
DL Toolbox	Matlab	Easy and simple to use	CNN/DBN/SAE
Caffe	C++/Python/Matlab	Fast implementation and simple to use	CNN/MLP/RNN
DL4J	Java/Scala/Python	Distributed support	CNN/DBN/SAE
Theano	Python	Flexible for models	CNN/DBN/SAE/RNN
Torch	Lua/C++	Fast implementation and flexible for models	CNN/DBN/SAE/RNN
TensorFlow	C++/Python/Java/R	Flexible reinforced for models	CNN/DBN/SAE/RNN
MXNet	C++/Python/Julia/R	Distributed support	CNN/RNN/DBN
Keras	Python/R	Easy and simple to use	CNN/RNN/DBN

Table 5 The list of basics of various data mining and deep learning algorithms

Technique	The key ideas and basic structures
Data Mining	
Data Preprocessing	
Data Cleaning	Fill in missing values, smooth noise, identify or delete outliers, and resolve inconsistencies
Data Integration	Combine data from multiple sources, store them together and provide a unified view of data
Data Transformation	Replace pattern distribution or relationship with a specific form that is suitable for mining
Data Reduction	A reduced representation of the dataset which is much smaller but still close to preserving the integrity of the original data
Classification	
KNN	The unclassified data point is discovered and assigned to particular label according to the formerly known k nearest neighbor (KNN) points, and vote the determination of targeted object belonging
NB	A probabilistic statistical classifier where the naïve assumption is made that attributes or features are conditionally independent
DT	Split features to outcomes of targeted class values through logical conjunctions of those attributes with traditional top-down divide-and-conquer approach utilizing information entropy
SVM	Construct hyper-plane by implicitly kernel function mapping of the original input space to higher dimensional one in order to make the distance between two separated classes as maximal as possible
NN	Multiple layers of grouped and interconnected processing neurons with adjustable weighted links and nonlinear mapping
EM	A hybrid of various approaches with the hypothesis behind that many classification models are able to work collaboratively and get superior classification outcomes than single one only
Clustering	
HC	Gather data objects into tiny clusters via average-linkage, where those smaller clusters are categorized into larger clusters from button to up layer hierarchically
PRC	Organize and construct several segmentations of data, and each partition is a subgroup of original collection of data for every individual cluster measured by distance
DBC	The density of data points distribution within the given radius of neighbors is used for arbitrary shaped clustering and it is the criteria of cluster partitioning
GBC	Multiresolution grid structures of the space of objects are formed into independently quantized cells, wherein all calculations of clustering steps are implemented
MBC	The supposed model for each cluster is given firstly and replace the best fitting model for previous one
Deep Learning	
DNN	A hierarchy form consists of deep layers of input, multi hidden and output of a NN, processing input with feedforward and optimizing with backpropagation
CNN	Contain multiple convolution layers to capture feature maps from regional connectivity via the weighted filter, and polling layers to reduce data size, at last a fully connected NN is added as the classifier
SAE	A well-constructed model representation with the built encoder and decoder. The encoder transforms the input vector to a hidden layer and the decoder maps the hidden layer representation to the reconstructed input, which is regarded as the output result.
DBN	It is composed of multiple restricted Boltzmann machines (RBMs) as each single RBM contains a visual layer and a hidden layer, a two way direction link between two layers, where the visual layer simultaneously works as input and output multiplexing
RNN	Recurrent and cyclic connection unit is designed for handling sequential information therefore causing old data remembered implicitly in state vectors, hidden unit relies on the computation of current input and previous values stored in those state vectors, output is affected by both past and current inputs consequently

data and stacking machines are widely used because of their low cost and quick effect in the industry. Whereas, deep learning algorithms need comparatively big data to learn sufficient knowledge from it and get better insight perfectly. When there is a small dataset for training, deep learning methods always perform worse results, so large and big data is a necessary condition of deep learning. The situation is also true when it comes to the scenario of bioinformatics data analysis.

Another striking difference between those two learning strategies is hardware environment. As a normal sense, low hardware configuration on end machines are enough for conventional machine learning algorithms. On the contrary, it heavily relies on high performance end devices for deep learning algorithms because of the ability of GPUs for supporting deep learning quite well. During the working operations, there are various and massive inherent mathematical calculations such as matrix multiplication within deep learning and optimization of its algorithms, the essence of GPUs can just meet the requirements of deep learning tasks.

- **Feature Engineering**: The feature is a particular representation of an observed object for the purpose of measuring itself by informative, independent and discriminating characteristics. And in feature engineering, its main process is the workflow of converting domain knowledge into the emergence of created and extracted features to reduce the complexity of original data, pick up meaningful expressions of raw data and make underlying patterns intuitively visible to learning algorithms. The price of deploying feature engineering is the difficult and expensive cost according to resource, expertise and time.

In Machine learning, human resource is dedicated to identify, encode and label the large majority of applied features by experts manually in terms of domain knowledge and data type. For instance, features from a biomedical image can be composed of pixel values, shapes of an object, textures in the area, positions and orientations of interested part. Then all these features are fed into specific machine learning algorithms to be evaluated and classified, finally the performance of machine learning is decided by the accuracy of feature identification and extraction from raw ones. Comparatively, more high-level features are learned from bottom up representation of original data in deep learning. It can combine multiple possibilities of distinctive features in a certain feature level, and build on stacked feature layers hierarchically. After consecutive hierarchical identification of complex patterns in raw data, it then selects which of these features are responsible for carrying out the learning job. As the result, deep learning automatically finds out the features applied to classification importantly and makes itself an adaptive feature extractor for ubiquitous problem solving, which is a prominent step forward of traditional machine learning. Examples can be found in CNN of biomedical images, where low-level features like edges, lines, textures and orientations of the target lesion are obtained and high-level expression of a whole lesion area is produced based on those features in former layers.

- **Problem Solving Police**: The core thought when applying traditional machine learning algorithms to solve a learning task is generally concluded as divide and conquers method, i.e. breaking down the issue recursively into individual and small sub components that can be solved simply enough, and the ultimate solution to original problem is the aggregation of sub approaches to those divided questions. Typical case can be referred to decision tree model built on information entropy and its improved theories. Imagine the application of breast cancer detection task, the aim is to identify multiple lesion regions in mammography data to predict whether the patient is suffering from it. As a usual machine learning approach, it decomposes the learning process into lesion detection and illness recognition. Image segmentation methods may be employed to check the boundaries and outlines of region of interest through scanning all possible divided parts of raw image, and object recognition algorithms like Bayesian or SVM methods are then utilized to identify the distribution characteristics in that region and tell whether it matches with breast cancer. However, as mentioned before, if settings of network model are done, deep learning can process the issue via the manner of end-to-end policy. One does not need to care about the detailed intermediate procedures of entire problem solving workflow; the deep neural network will automatically generate the predictive result when directly passing the image as input. Their linkage is that they both use self-learning method to solve problems as training and testing data is given, and the most areas of problem solving basically overlap, i.e. their targets are the same.
- **Interpretability**: The primitive deep learning is a learning process that uses deep neural network to solve feature expressions and can be roughly regarded as a neural network structure that contains multiple hidden layers. As far as machine learning is concerned, models generated from corresponding algorithms are with many forms but relatively simple, which leads to an easily interpretable structure of model itself such as decision tree, rule based reasoning and logistic regression. But for deep learning, there are only several commonly used algorithms and network architectures, but both are very complicated to design and explain in order to get a comprehensive understanding of all. The main problem of traditional machine learning is finding suitable models; meanwhile the primary mission tends to search proper parameters of models for deep learning since more attentions are paid on trying to boost the interpretability of deep network structure. Indeed mathematically one can find out which nodes of a deep neural network are activated, but still cannot know exactly what these neurons are supposed to model or what these layers of neurons are doing collectively.

Challenges and issues

While machine learning and deep learning have illustrated their powerful abilities and strengths dealing with bioinformatics data processing in preprocessing, feature extraction, feature selection, classification, clustering, etc. They still face some kinds of inherent challenges and emerging issues that should be taken into considerations.

Data preprocessing and imbalanced data Although the topic of data preprocessing is discussed before in the relevant section, the properties of bioinformatics data with high and complex dimensional features, tremendous amount of instances, heterogeneous from multiple sources make the procedure quite tough as time goes by and new problem emerges [113]. For example, it is really challenging to fuse informative biological data together with classic e-records for patients or medical images or other types of data in a commercial information system that only has limited data fusion functions. The similar question appears under the situation of merging multiple datasets when concerning the criteria both in scale and accuracy. In spite of the well-known curse of dimensionality, a further problem of vast biomedical data has the essence of incompleteness where the raw data involves missing data values, inconsistent value naming conventions or the detection and removal of duplicate entries are needed.

Unfortunately for imbalanced data issue in bioinformatics [114], the reality is that the expensive data acquisition stage and complex processing situation have restricted the size of data thus prominent asymmetric label distribution is shown (for instance in an illness center database, healthy cases are always significantly less than those sick ones because there is no need for a normal man to go to the hospital). Some researchers have been dedicating in solving imbalanced data problem using various sampling or feature extraction methods, cost sensitive learning, pre-training etc.

Big data needs for large datasets Over the years advanced technologies have facilitated the emergence of new ways of generating and collecting data rapidly, continuously and largely, namely big data with properties characterized by 5Vs: volume (large data quantity), variety (inconsistent data type and nature), velocity (fast data speed), variability or veracity (data inconsistency) and value (ultimate target and goal) [115–122]. Indeed it is also true for the application of bioinformatics field, and it leads to lots of issues to be addressed both in academic research and industrial production. Real time analytics on big data is much harder than before due to the 5Vscharacteristics of big data, usually batch-based data analytics mode is expanded as distributed and parallel computing algorithms to cope with velocity issue [123–127]. But the capability of data throughput of I/O operations is a bottleneck of real time analytics performance [116]. And also most of the AI learning algorithms are not primitively created for distributed and parallel computation, even though there are some frameworks that build deep learning on big data platform like Tensorflow on Spark, they are still under incubation or research oriented with the shortage of robust industrial implementation.

Learning strategy and model selection In data mining algorithm domain, except for well-represented data structure, it has struggled against not only the big amount of unstructured and heterogeneous data, but also data obtaining massive interrelated objects whose original formation is in point clouds or typed graphs [113, 117]. So more advanced learning strategies are desired for novel insights for pattern discovery. Firstly, graph theory is vigorous method to map interrelated objects with types of structures. The hybrid point of graph-based learning approach can be found in aspects oftopology, network analysis, data conceptual representation, etc. The second is topology-based learning method, which has the potential power on arbitrarily high-dimensional data from computational geometry. Till now it is hardly implemented to the extent of human brain pattern recognition. Generally speaking, for the third entropy-based learning strategy, the deserved problem is to apply more intricate information theories in machine learning algorithms.

In deep learning model architecture, the interpretability and quality control is a major criticism against the intuition known as black-box, which means that the hidden and internal structure keeps unclear and difficult to explain even if the results are awesome. And the situation is always worse in bioinformatics context. Visualization is treated as a widespread solution of black-box, however the deep model is still suffering from a lack of transparency to discover complex structural relationships in bioinformatics.

Another widely talked topic locates in the choice of appropriate models under different circumstances. One can refer to Table 2 and 4 in previous sections and find the pros and cons of each model, so in order to get the reliable and robust consequences of data analytics, the selection of various models with capabilities to handle input data and learning target. While in reality, the understandings about model architectures especially for DNNs are very shallow and rough. The hyperparameter optimization task is not straightforward all the time after model selection step, such as the learning rates, initial weight and bias values, number of hidden layers and neural cells, iteration steps and so on. Hyperparameter optimization is directly related to accelerating deep learning and computational cost reduction problems. Many scholars have contributed a lot in this study area, nevertheless, the increasing demand of improvements is worthy all the time.

Conclusion

The rapid growth and development of bioinformatics has created an unprecedented opportunity for biomedical research, data analytics, knowledge discovery and innovative application. Data mining with core techniques of machine learning algorithms is a significantly promising approach to achieve such goals in both explicit and implicit IT ways. However, some poor issues such as shallow, static, batch oriented and non-generalized performance emerge, causing troublesome bottleneck. Nevertheless, the emergence of deep learning extends the boundary of conventional machine learning with more generalized ability to overcome challenges. Through this review paper, the importance of both biomedicine and bioinformatics is described, following with the basics of data mining and deep learning methodologies, among which multiple data preprocessing, classification and clustering algorithms are discussed of machine learning and various deep frameworks are analyzed in bioinformatics application scenario. As a result, it is strongly believed that deep learning together with machine learning are anticipated to make great advances collaboratively in development of biomedicine and bioinformatics perspectives.

For the future of this study, we think that the aggregation of advanced machine learning algorithms can make the promising progress of technical improvement, and data fusion method and efficiency assessment of bioinformatics data will also contribute in the direction of present applications. Software design and tools development in the manner of automated way can significantly boost machine learning evolution to a certain degree. And in deep learning domain, a bright future trend is to joint traditional deep architectures and try a better integration with higher performance. Furthermore by extension of traditional deep methods, semi-supervised learning, reinforcement learning, transfer learning, etc. are acquiring more and more attentions particularly with the incorporation of big data processing techniques.

Funding The authors are thankful to the financial support from the research grants, 1) MYRG2015–00024-FST, titled Building Sustainable Knowledge Networks through Online Communities' offered by RDAO/FST, University of Macau and Macau SAR government. 2) MYRG2016–00069, titled 'Nature-Inspired Computing and Metaheuristics Algorithms for Optimizing Data Mining Performance' offered by RDAO/FST, University of Macau and Macau SAR government. 3) FDCT/126/2014/A3, titled 'A Scalable Data Stream Mining Methodology: Stream-based Holistic Analytics and Reasoning in Parallel' offered by FDCT of Macau SAR government.

Compliance with Ethical Standards

Conflict of Interest The authors declare that this article content has no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Li, J., Wong, L., and Yang, Q., Guest editors' introduction: Data Mining in Bioinformatics. IEEE Intell. Syst. 20(6):16–18, 2005.
2. Yoo, I., Alafaireet, P., Marinov, M., Pena-Hernandez, K., Gopidi, R., Chang, J.-F., and Hua, L., Data mining in healthcare and biomedicine: a survey of the literature. J. Med. Syst. 36(4):2431–2448, 2012.
3. Kharya, S., Using data mining techniques for diagnosis and prognosis of cancer disease. arXiv preprint arXiv:12051923, 2012.
4. Santosh, K., and Antani, S., Automated chest X-ray screening: Can lung region symmetry help detect pulmonary abnormalities? IEEE Transactions on Medical Imaging, 2017.
5. Zohora, F. T., Antani, S., and Santosh, K., Circle-like foreign element detection in chest x-rays using normalized cross-correlation and unsupervised clustering. In: Medical Imaging 2018: Image Processing. International Society for Optics and Photonics, p 105741V, 2018.
6. Zohora, F. T., and Santosh, K., Foreign Circular Element Detection in Chest X-Rays for Effective Automated Pulmonary Abnormality Screening. International Journal of Computer Vision and Image Processing (IJCVIP). 7(2):36–49, 2017.
7. Santosh, K., Vajda, S., Antani, S., and Thoma, G. R., Edge map analysis in chest X-rays for automatic pulmonary abnormality screening. Int. J. Comput. Assist. Radiol. Surg. 11(9):1637–1646, 2016.
8. Karargyris, A., Siegelman, J., Tzortzis, D., Jaeger, S., Candemir, S., Xue, Z., Santosh, K., Vajda, S., Antani, S., and Folio, L., Combination of texture and shape features to detect pulmonary abnormalities in digital chest X-rays. Int. J. Comput. Assist. Radiol. Surg. 11(1):99–106, 2016.
9. Kalsi, S., Kaur, H., and Chang, V., DNA Cryptography and Deep Learning using Genetic Algorithm with NW algorithm for Key Generation. J. Med. Syst. 42(1):17, 2018.
10. Hsieh, S.-L., Hsieh, S.-H., Cheng, P.-H., Chen, C.-H., Hsu, K.-P., Lee, I.-S., Wang, Z., and Lai, F., Design ensemble machine learning model for breast cancer diagnosis. J. Med. Syst. 36(5):2841–2847, 2012.
11. Somasundaram, S., Alli, P., and Machine Learning, A., Ensemble Classifier for Early Prediction of Diabetic Retinopathy. J. Med. Syst. 41(12):201, 2017.
12. Alanazi, H. O., Abdullah, A. H., and Qureshi, K. N., A critical review for developing accurate and dynamic predictive models using machine learning methods in medicine and health care. J. Med. Syst. 41(4):69, 2017.
13. Han, J., How can data mining help bio-data analysis? In: Proceedings of the 2nd International Conference on Data Mining in Bioinformatics. Springer-Verlag, pp 1–2, 2002.
14. Brazma, A., Hingamp, P., Quackenbush, J., Sherlock, G., Spellman, P., Stoeckert, C., Aach, J., Ansorge, W., Ball, C. A., and Causton, H. C., Minimum information about a microarray experiment (MIAME)—toward standards for microarray data. Nat. Genet. 29(4):365–371, 2001.
15. Antonie, M.-L., Zaiane, O. R., and Coman, A. Application of data mining techniques for medical image classification. In: Proceedings of the Second International Conference on Multimedia Data Mining. Springer-Verlag, pp. 94–101, 2001.
16. Dasu, T., Johnson, T., Muthukrishnan, S., and Shkapenyuk, V., Mining database structure; or, how to build a data quality browser. In: Proceedings of the 2002 ACM SIGMOD international conference on Management of data. ACM, pp 240–251, 2002.

17. Raman, V., and Hellerstein, J. M., Potter's wheel: An interactive data cleaning system. In: VLDB, pp 381–390, 2001.
18. Becker, B., Kohavi, R., and Sommerfield, D., Visualizing the simple Bayesian classifier. Information Visualization in Data Mining and Knowledge Discovery. 18:237–249, 2001.
19. Zhang, J., Hsu, W., and Lee, M., FASTCiD: FAST clustering in dynamic spatial databases. Submitted for publication, 2002.
20. Xu, X., Jäger, J., and Kriegel, H.-P., A fast parallel clustering algorithm for large spatial databases. In: High Performance Data Mining. Springer, pp 263–290, 1999.
21. Han, J., Pei, J., and Kamber, M., Data mining: concepts and techniques. New York: Elsevier, 2011.
22. Daubechies, I., Ten lectures on wavelets. SIAM, 1992.
23. Mackiewicz, A., and Ratajczak, W., Principal components analysis (PCA). Comput. Geosci. 19:303–342, 1993.
24. Holland, S. M., Principal components analysis (PCA). Department of Geology. Athens, GA: University of Georgia, 2008, 30602–32501.
25. Ku, W., Storer, R. H., and Georgakis, C., Disturbance detection and isolation by dynamic principal component analysis. Chemom. Intell. Lab. Syst. 30(1):179–196, 1995.
26. Andrews, H., and Patterson, C., Singular value decomposition (SVD) image coding. IEEE Trans. Commun. 24(4):425–432, 1976.
27. Shearer, C., The CRISP-DM model: the new blueprint for data mining. Journal of Data Warehousing 5(4):13–22, 2000.
28. Glas, A. M., Floore, A., Delahaye, L. J., Witteveen, A. T., Pover, R. C., Bakx, N., Lahti-Domenici, J. S., Bruinsma, T. J., Warmoes, M. O., and Bernards, R., Converting a breast cancer microarray signature into a high-throughput diagnostic test. BMC Genomics 7(1):278, 2006.
29. Yoshida, H., Kawaguchi, A., and Tsuruya, K., Radial basis function-sparse partial least squares for application to brain imaging data. Computational and Mathematical Methods in Medicine 2013, 2013.
30. Jen, C.-H., Wang, C.-C., Jiang, B. C., Chu, Y.-H., and Chen, M.-S., Application of classification techniques on development an early-warning system for chronic illnesses. Expert Syst. Appl. 39(10):8852–8858, 2012.
31. Bailey, T., and Jain, A., A note on distance-weighted $ k $-nearest neighbor rules. IEEE Trans Syst Man Cybern 4:311–313, 1978.
32. Keller, J. M., Gray, M. R., and Givens, J. A., A fuzzy k-nearest neighbor algorithm. IEEE Trans Syst Man Cybern 4:580–585, 1985.
33. Liu, D.-Y., Chen, H.-L., Yang, B., Lv, X.-E., Li, L.-N., and Liu, J., Design of an enhanced fuzzy k-nearest neighbor classifier based computer aided diagnostic system for thyroid disease. J. Med. Syst. 36(5):3243–3254, 2012.
34. Syaliman, K., and Nababan, E., Sitompul O Improving the accuracy of k-nearest neighbor using local mean based and distance weight. In: Journal of Physics: Conference Series. vol 1. IOP Publishing, p 012047, 2018.
35. Spiegelhalter, D. J., Dawid, A. P., Lauritzen, S. L., Cowell, R. G., Bayesian analysis in expert systems. Statistical science: 219–247, 1993.
36. Kononenko, I., Semi-naive Bayesian classifier. In: Machine Learning—EWSL-91. Springer, pp 206–219, 1991.
37. Langley, P., Induction of recursive Bayesian classifiers. In: Machine Learning: ECML-93. Springer, pp 153–164, 1993.
38. Peng, H., and Long, F. A., Bayesian learning algorithm of discrete variables for automatically mining irregular features of pattern images. In: Proceedings of the Second International Conference on Multimedia Data Mining. Springer-Verlag, pp 87–93, 2001.
39. Hickey, S. J., Naive Bayes classification of public health data with greedy feature selection. Commun. IIMA 13(2):7, 2013.
40. Abellán, J., and Castellano, J. G., Improving the Naive Bayes Classifier via a Quick Variable Selection Method Using Maximum of Entropy. Entropy 19(6):247, 2017.
41. Estella, F., Delgado-Marquez, B. L., Rojas, P., Valenzuela, O., San Roman, B., and Rojas, I., Advanced system for automously classify brain MRI in neurodegenerative disease. In: Multimedia Computing and Systems (ICMCS), 2012 International Conference on. IEEE, pp 250–255, 2012.
42. Rodriguez, J. J., Kuncheva, L. I., and Alonso, C. J., Rotation forest: A new classifier ensemble method. IEEE Trans. Pattern Anal. Mach. Intell. 28(10):1619–1630, 2006.
43. Domingos, P., and Hulten, G., Mining high-speed data streams. In: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp 71–80, 2000.
44. Hulten, G., Spencer, L., and Domingos, P., Mining time-changing data streams. In: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp 97–106, 2001.
45. Zhu, B., Jiao, J., Han, Y., Weissman, T., Improving Decision Tree Learning by Optimal Split Scoring Function Estimation, 2017.
46. Esmeir, S., and Markovitch, S., Anytime induction of low-cost, low-error classifiers: a sampling-based approach. J. Artif. Intell. Res. 33:1–31, 2008.
47. Esmeir, S., and Markovitch, S., Anytime learning of anycost classifiers. Mach. Learn. 82(3):445–473, 2011.
48. Boser, B. E., Guyon, I. M., and Vapnik, V. N. A., training algorithm for optimal margin classifiers. In: Proceedings of the fifth annual workshop on Computational learning theory. ACM, pp 144–152, 1992.
49. Lee, K.-J., Hwang, Y.-S., and Rim, H.-C., Two-phase biomedical NE recognition based on SVMs. In: Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine-Volume 13. Association for Computational Linguistics, pp 33–40, 2003.
50. Nanni, L., Lumini, A., and Brahnam, S., Survey on LBP based texture descriptors for image classification. Expert Syst. Appl. 39(3):3634–3641, 2012.
51. Hasri, N. N. M., Wen, N. H., Howe, C. W., Mohamad, M. S., Deris, S., and Kasim, S., Improved Support Vector Machine Using Multiple SVM-RFE for Cancer Classification. International Journal on Advanced Science, Engineering and Information. Technology 7(4–2):1589–1594, 2017.
52. Kavitha, K., and Gopinath, A., Gopi M Applying improved svm classifier for leukemia cancer classification using FCBF. In: Advances in Computing, Coemmunications and Informatics (ICACCI), 2017 International Conference on. IEEE, pp 61–66, 2017.
53. Er, O., Yumusak, N., and Temurtas, F., Chest diseases diagnosis using artificial neural networks. Expert Syst. Appl. 37(12):7648–7655, 2010.
54. Gunasundari, S., and Baskar S., Application of Artificial Neural Network in identification of lung diseases. In: Nature & Biologically Inspired Computing. NaBIC 2009. World Congress on. IEEE, pp 1441–1444, 2009.
55. Bin, W., and Jing, Z., A novel artificial neural network and an improved particle swarm optimization used in splice site prediction. J Appl Computat Math 3(166), 2014.
56. Amaratunga, D., Cabrera, J., and Lee, Y.-S., Enriched random forests. Bioinformatics 24(18):2010–2014, 2008.
57. Yao, D., Yang, J., and Zhan, X., An improved random forest algorithm for class-imbalanced data classification and its application in PAD risk factors analysis. Open Electr Electron Eng J 7(1): 62–72, 2013.
58. Fabris, F., Doherty, A., Palmer, D., de Magalhães, J. P., Freitas, A. A., and Wren, J., A new approach for interpreting Random Forest models and its application to the biology of ageing. Bioinformatics 1:8, 2018.

59. Gopal, R., Marsden, J. R., and Vanthienen, J., Information mining—Reflections on recent advancements and the road ahead in data, text, and media mining. New York, NY: Elsevier, 2011.
60. Ding, J., Berleant, D., Nettleton, D., and Wurtele, E., Mining MEDLINE: abstracts, sentences, or phrases. In: Proceedings of the pacific symposium on biocomputing, 2002. pp 326–337, 2002.
61. Shen, H.-B., and Chou, K.-C., Ensemble classifier for protein fold pattern recognition. Bioinformatics 22(14):1717–1722, 2006.
62. Eom, J.-H., Kim, S.-C., and Zhang, B.-T., AptaCDSS-E: A classifier ensemble-based clinical decision support system for cardiovascular disease level prediction. Expert Syst. Appl. 34(4):2465–2479, 2008.
63. Jain, A. K., Murty, M. N., and Flynn, P. J., Data clustering: a review. ACM computing surveys (CSUR) 31(3):264–323, 1999.
64. Zhang, T., Ramakrishnan, R., and Livny, M., BIRCH: an efficient data clustering method for very large databases. In: ACM Sigmod Record. vol 2. ACM, pp 103–114, 1996.
65. Bryant, D., and Moulton, V., Neighbor-net: an agglomerative method for the construction of phylogenetic networks. Mol. Biol. Evol. 21(2):255–265, 2004.
66. Heo, M., and Leon, A. C., Statistical power and sample size requirements for three level hierarchical cluster randomized trials. Biometrics 64(4):1256–1262, 2008.
67. Darkins, R., Cooke, E. J., Ghahramani, Z., Kirk, P. D., Wild, D. L., and Savage, R. S., Accelerating Bayesian hierarchical clustering of time series data with a randomised algorithm. PLoS One 8(4): e59795, 2013.
68. Elkamel, A., Gzara, M., and Ben-Abdallah, H., A bio-inspired hierarchical clustering algorithm with backtracking strategy. Appl. Intell. 42(2):174–194, 2015.
69. Yildirim, P., and Birant, D., K-Linkage: A New Agglomerative Approach for Hierarchical Clustering. Adv Electr Comput Eng 17(4):77–88, 2017.
70. Chiu, T., Fang, D., Chen, J., Wang, Y., and Jeris, C., A robust and scalable clustering algorithm for mixed type attributes in large database environment. In: Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 263–268, 2001.
71. Hussain, H. M., Benkrid, K., Seker, H., and Erdogan, A. T., FPGA implementation of K-means algorithm for bioinformatics application: An accelerated approach to clustering Microarray data. In: Adaptive Hardware and Systems (AHS), 2011 NASA/ESA Conference on. IEEE, pp 248–255, 2011.
72. Tseng, G. C., Penalized and weighted K-means for clustering with scattered objects and prior information in high-throughput biological data. Bioinformatics 23(17):2247–2255, 2007.
73. Botía, J. A., Vandrovcova, J., Forabosco, P., Guelfi, S., D'Sa, K., Hardy, J., Lewis, C. M., Ryten, M., and Weale, M. E., An additional k-means clustering step improves the biological features of WGCNA gene co-expression networks. BMC Syst. Biol. 11(1): 47, 2017.
74. Sathiya, G., and Kavitha, P., An efficient enhanced K-means approach with improved initial cluster centers. Middle-East J. Sci. Res. 20(1):100–107, 2014.
75. Jain, A. K., Data clustering: 50 years beyond K-means. Pattern Recogn. Lett. 31(8):651–666, 2010.
76. Jiang, D., Pei, J., and Zhang, A., DHC: a density-based hierarchical clustering method for time series gene expression data. In: Bioinformatics and Bioengineering. Proceedings. Third IEEE Symposium on, 2003. IEEE, pp 393–400, 2003.
77. Kailing, K., Kriegel, H.-P., and Kröger, P., Density-connected subspace clustering for high-dimensional data. In: Proceedings of the 2004 SIAM International Conference on Data Mining. SIAM, pp 246–256, 2004.
78. Wang, L., Li, M., Han, X., and Zheng, K., An improved density-based spatial clustering of application with noise. International Journal of Computers and Applications: 1–7, 2018.
79. Günnemann, S., Boden, B., and Seidl, T., DB-CSC: a density-based approach for subspace clustering in graphs with feature vectors. Machine Learning and Knowledge Discovery in Databases:565–580, 2011.
80. Sittel, F., and Stock, G., Robust density-based clustering to identify metastable conformational states of proteins. J. Chem. Theory Comput. 12(5):2426–2435, 2016.
81. Liu, S., Zhu, L., Sheong, F. K., Wang, W., and Huang, X., Adaptive partitioning by local density-peaks: An efficient density-based clustering algorithm for analyzing molecular dynamics trajectories. J. Comput. Chem. 38(3):152–160, 2017.
82. Maltsev, N., Glass, E., Sulakhe, D., Rodriguez, A., Syed, M. H., Bompada, T., Zhang, Y., and D'souza, M., PUMA2—grid-based high-throughput analysis of genomes and metabolic pathways. Nucleic Acids Res. 34(suppl_1):D369–D372, 2006.
83. Ortuso, F., Langer, T., and Alcaro, S., GBPM: GRID-based pharmacophore model: concept and application studies to protein–protein recognition. Bioinformatics 22(12):1449–1455, 2006.
84. Porro, I., Torterolo, L., Corradi, L., Fato, M., Papadimitropoulos, A., Scaglione, S., Schenone, A., and Viti, F., A Grid-based solution for management and analysis of microarrays in distributed experiments. BMC Bioinf 8(1):S7, 2007.
85. Ren, J., Cai, B., and Hu, C., Clustering over data streams based on grid density and index tree. 6. https://doi.org/10.4156/jcit.vol6.issue1.11, 2011.
86. Liu, F., Ye, C., and Zhu, E., Accurate Grid-based Clustering Algorithm with Diagonal Grid Searching and Merging. In: IOP Conference Series: Materials Science and Engineering. 1: IOP Publishing, p 012123, 2017.
87. Si, Y., Liu, P., Li, P., and Brutnell, T. P., Model-based clustering for RNA-seq data. Bioinformatics 30(2):197–205, 2013.
88. Abawajy, J. H., Kelarev, A. V., and Chowdhury, M., Multistage approach for clustering and classification of ECG data. Comput. Methods Prog. Biomed. 112(3):720–730, 2013.
89. Wang, J., Delabie, J., Aasheim, H. C., Smeland, E., and Myklebost, O., Clustering of the SOM easily reveals distinct gene expression patterns: results of a reanalysis of lymphoma study. BMC Bioinf 3(1):36, 2002. https://doi.org/10.1186/1471-2105-3-36.
90. Hinton, G. E., and Salakhutdinov, R. R., Reducing the dimensionality of data with neural networks. Science 313(5786):504–507, 2006.
91. Hinton, G. E., Osindero, S., and Teh, Y.-W., A fast learning algorithm for deep belief nets. Neural Comput. 18(7):1527–1554, 2006.
92. Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H., Greedy layer-wise training of deep networks. In: Advances in neural information processing systems. pp 153–160, 2007.
93. LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P., Gradient-based learning applied to document recognition. Proc. IEEE 86(11): 2278–2324, 1998.
94. Pascanu, R., Mikolov, T., and Bengio, Y., On the difficulty of training recurrent neural networks. In: International Conference on Machine Learning. pp 1310–1318, 2013.
95. Hubel, D. H., and Wiesel, T. N., Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J. Physiol. 160(1):106–154, 1962.
96. Xu, J., Xiang, L., Hang, R., and Wu, J., Stacked Sparse Autoencoder (SSAE) based framework for nuclei patch classification on breast cancer histopathology. In: Biomedical Imaging (ISBI), 2014 IEEE 11th International Symposium on. IEEE, pp 999–1002, 2014.

97. Jia, W., Yang, M., and Wang, S.-H., Three-Category Classification of Magnetic Resonance Hearing Loss Images Based on Deep Autoencoder. J. Med. Syst. 41(10):165, 2017.

98. Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A., Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on Machine learning. ACM, pp 1096–1103, 2008.

99. Huang, G. B., Lee, H., and Learned-Miller, E., Learning hierarchical representations for face verification with convolutional deep belief networks. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, pp 2518–2525. , 2012.

100. Lee, H., Pham, P., Largman, Y., Ng AY., Unsupervised feature learning for audio classification using convolutional deep belief networks. In: Advances in neural information processing systems, 2009. pp 1096–1104, 2009.

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

102. Bengio, Y., Simard, P., and Frasconi, P., Learning long-term dependencies with gradient descent is difficult. IEEE Trans. Neural Netw. 5(2):157–166, 1994.

103. Gers, F. A., Schmidhuber, J., and Cummins F., Learning to forget: Continual prediction with LSTM.

104. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio Y., Learnieng phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:14061078

105. Fakoor, R., Ladhak, F., Nazi, A., and Huber, M., Using deep learning to enhance cancer diagnosis and classification. In: Proceedings of the International Conference on Machine Learning, 2013.

106. Liang, M., Li, Z., Chen, T., and Zeng, J., Integrative data analysis of multi-platform cancer data with a multimodal deep learning approach. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 12(4):928–937, 2015.

107. Gao, X., Lin, S., and Wong, T. Y., Automatic feature learning to grade nuclear cataracts based on deep learning. IEEE Trans. Biomed. Eng. 62(11):2693–2701, 2015.

108. Liao, S., Gao, Y., Oto, A., and Shen, D., Representation learning: a unified deep learning framework for automatic prostate MR segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2013. Springer, pp 254–261, 2013.

109. Di Lena, P., Nagata, K., and Baldi, P., Deep architectures for protein contact map prediction. Bioinformatics 28(19):2449–2457, 2012.

110. Ditzler, G., Polikar, R., and Rosen, G., Multi-layer and recursive neural networks for metagenomic classification. IEEE Trans on Nanobiosci 14(6):608–616, 2015.

111. Majumdar, A., Real-time Dynamic MRI Reconstruction using Stacked Denoising Autoencoder. arXiv preprint arXiv: 150306383, 2015.

112. Xu, Y., Dai, Z., Chen, F., Gao, S., Pei, J., and Lai, L., Deep learning for drug-induced liver injury. J. Chem. Inf. Model. 55(10): 2085–2093, 2015.

113. Holzinger, A., Dehmer, M., and Jurisica, I., Knowledge discovery and interactive data mining in bioinformatics-state-of-the-art, future challenges and research directions. BMC Bioinf 15(6):I1, 2014.

114. Min, S., Lee, B., and Yoon, S., Deep learning in bioinformatics. Brief. Bioinform. 18(5):851–869, 2017.

115. Lan, K., Fong, S., Song, W., Vasilakos, A. V., and Millham, R. C., Self-Adaptive Pre-Processing Methodology for Big Data Stream Mining in Internet of Things Environmental Sensor Monitoring. Symmetry 9(10):244, 2017.

116. Kashyap, H., Ahmed, H. A., Hoque, N., Roy, S., and Bhattacharyya, D. K., Big data analytics in bioinformatics: A machine learning perspective. arXiv preprint arXiv:150605101, 2015.

117. Holzinger, A., and Jurisica I., Knowledge discovery and data mining in biomedical informatics: The future is in integrative, interactive machine learning solutions. In: Interactive knowledge discovery and data mining in biomedical informatics. Springer, pp 1–18, 2014.

118. Kamal, S., Ripon, S. H., Dey, N., Ashour, A. S., and Santhi, V., A MapReduce approach to diminish imbalance parameters for big deoxyribonucleic acid dataset. Comput. Methods Prog. Biomed. 131:191–206, 2016.

119. Bhatt, C., Dey, N., and Ashour, A. S., Internet of things and big data technologies for next generation healthcare, 2017.

120. Dey, N., Hassanien, A. E., Bhatt, C., Ashour, A., and Satapathy, S. C., Internet of Things and Big Data Analytics Toward Next-Generation Intelligence. Berlin: Springer, 2018.

121. Tamane, S., Tamane, S., Solanki, V. K., and Dey, N., Privacy and security policies in big data, 2017.

122. Dey, N., Bhatt, C., and Ashour, A. S., Big Data for Remote Sensing: Visualization, Analysis and Interpretation, 2018.

123. Kamal, M. S., Dey, N., and Ashour, A. S., Large Scale Medical Data Mining for Accurate Diagnosis: A Blueprint. In Handbook of Large-Scale Distributed Computing in Smart Healthcare (pp. 157–176). Springer: Cham, 2017.

124. Manogaran, G., and Lopez, D., Disease surveillance system for big climate data processing and dengue transmission. International Journal of Ambient Computing and Intelligence (IJACI) 8(2):88–105, 2017.

125. Jain, A., and Bhatnagar, V., Concoction of Ambient Intelligence and Big Data for Better Patient Ministration Services. International Journal of Ambient Computing and Intelligence (IJACI) 8(4):19–30, 2017.

126. Matallah, H., Belalem, G., and Bouamrane, K., Towards a New Model of Storage and Access to Data in Big Data and Cloud Computing. International Journal of Ambient Computing and Intelligence (IJACI) 8(4):31–44, 2017.

127. Vengadeswaran, S., and Balasundaram, S. R., An Optimal Data Placement Strategy for Improving System Performance of Massive Data Applications Using Graph Clustering. International Journal of Ambient Computing and Intelligence (IJACI) 9(3):15–30, 2018.