Accepted Manuscript

Chensi Cao, Feng Liu, Hai Tan, Deshou Song, Wenjie Shu, Weizhong Li, Yiming Zhou, Xiaochen Bo, Zhi Xie

Deep Learning and its Applications in Biomedicine

Chensi Cao[1, #,a], Feng Liu[2,#,b], Hai Tan[3,#,c], Deshou Song[3,d], Wenjie Shu[2,e], Weizhong Li[4,f], Yiming Zhou[1,5,*,g], Xiaochen Bo[2,*,h], Zhi Xie[3,*,i]

[1]*CapitalBio Corporation, Beijing 102206, China*

[2]*Department of Biotechnology, Beijing Institute of Radiation Medicine, Beijing 100850, China*

[3]*State Key Lab of Ophthalmology, Zhongshan Ophthalmic Center, Sun Yat-sen University, Guangzhou 500040, China*

[4]*Zhongshan School of Medicine, Sun Yat-sen University, Guangzhou 500040, China*

[5]*Department of Biomedical Engineering, Medical Systems Biology Research Center, Tsinghua University School of Medicine, Beijing 100084, China*

[#] Equal contribution.

[*] Corresponding authors.

E-mail: yimingzhou@capitalbio.com (Zhou Y); boxc@bmi.ac.cn (Bo X); xiezhi@gmail.com (Xie Z).

Running title: *Cao C et al / Deep Learning for Biomedicine*

[a]ORCID: 0000-0002-5262-6959.

[b]ORCID: 0000-0001-9410-676X.

[c]ORCID: 0000-0002-0616-5645

[d]ORCID: 0000-0001-5989-1635.

[e]ORCID: 0000-0001-9044-2352.

[f]ORCID:0000-0002-9003-7733 .

[g]ORCID: 0000-0001-7103-1814.

^hORCID: 0000-0003-3490-5812.

ⁱORCID: 0000-0002-5589-4836.

Total count of word: 6215

Total count of figures: 4

Total count of tables: 1

Abstract

Advances in biological and medical technologies have been providing us explosive volumes of biological and physiological data, such as medical images, electroencephalography, genomic and protein sequences. Learning from these data facilitates the understanding of human health and disease. Developed from artificial neural networks, deep learning-based algorithms show great promise in extracting features and learning patterns from complex data. The aim of this paper is to provide an overview of deep learning techniques and some of the state-of-the-art applications in the biomedical field. We first introduce the development of artificial neural network and deep learning. We then describe two main components of deep learning, *i.e.*, deep learning architectures and model optimization. Subsequently, some examples are demonstrated for deep learning applications, including medical image classification, genomic sequence analysis, as well as protein structure classification and prediction. Finally, we offer our perspectives for the future directions in the field of deep learning.

KEYWORDS: Deep learning; Big data; Bioinformatics; Biomedical informatics; Medical image; High-throughput sequencing

Introduction

Deep learning is a recent and fast-growing field of machine learning. It attempts to model abstraction from large-scale data by employing multi-layered deep neural networks (DNNs), thus making sense of data such as images, sounds, and texts [1]. Deep learning in general has two properties: (1) multiple layers of nonlinear processing units, and (2) supervised or unsupervised learning of feature presentations on each layer [1]. The early framework for deep learning was built on artificial neural networks (ANNs) in the 1980s [2], while the real impact of deep learning became apparent in 2006 [3,4]. Since then, deep learning has been applied to a wide range of fields, including automatic speech recognition, image recognition, natural language processing, drug discovery, and bioinformatics [5–7].

The past decades have witnessed a massive growth in biomedical data, such as genomic sequences, protein structures, and medical images, due to the advances of high-throughput technologies. This deluge of biomedical big data necessitates effective and efficient computational tools to store, analyze, and interpret such data [5,8]. Deep learning-based algorithmic frameworks shed light on these challenging problems. The aim of this paper is to provide the bioinformatics and biomedical informatics community an overview of deep learning techniques and some of the state-of-the-art applications of deep learning in the biomedical field. We hope this paper will provide readers an overview of deep learning, and how it can be used for analyzing biomedical data.

The development of ANNs

As a basis for deep learning, ANNs were inspired by biological processes in the 1960s, when it was discovered that different visual cortex cells were activated when cats visualized different objects [9,10]. These studies illustrated that there were connections between the eyes and the cells of the visual cortex, and that the information was processed layer by layer in the visual system. ANNs mimicked the perception of objects by connecting artificial neurons within layers that could extract the features of objects [11–16]. However, ANN research stagnated after the 1960s, due to the low capability resulting from its shallow structures and the limited computational capacity of computers at that time [17].

Thanks to the improvement in computer capabilities and methodologies [18], ANNs with efficient backpropagation (BP) facilitated studies on pattern recognition [19–23]. In a neural network with BP, classifications were first processed by the ANN model, and weights were then

modified by evaluating the difference between the predicted and the true class labels. Although BP helped to minimize errors through gradient descent, it seemed to work only for certain types of ANNs [24]. Through improving the steeper gradients with BP, several learning methods were proposed, such as momentum [25], adaptive learning rate [26–28], least-squares methods [29,30], quasi-Newton methods [31–34], and conjugate gradient (CG) [35,36]. However, due to the complexity of ANNs, other simple machine learning algorithms, such as support vector machines (SVMs) [37], random forest [38,39], and k-nearest neighbors algorithms (k-NN) [40], gradually overtook ANNs in popularity (**Figure 1**).

The development of deep learning

An ANN with more hidden layers offers much higher capacity for feature extraction [4]. However, an ANN often converges to the local optimum, or encounters gradient diffusion when it contains deep and complex structures [41]. A gradient propagated backwards rapidly diminishes in magnitude along the layers, resulting in slight modification to the weights in the layers near the input (http://deeplearning.stanford.edu/wiki/index.php/UFLDL_Tutorial) [42]. Subsequently, a layer-wise pre-training deep auto-encoder (AE) network was proposed, bringing ANNs to a new stage of development [3,4,43−45] (Figure 1). In this network, each layer is trained by minimizing the discrepancy between the original and the reconstructed data [4]. The layer-wise pre-training breaks the barrier of gradient diffusion [4], and also results in a better choice of weights for deep neural networks (DNNs), thereby preventing the reconstructed data from reaching a local optimum where the local optimum is usually caused by the random selection of initial weights. In addition, the employment of graphic processing units (GPUs) also renews the interest of researchers in deep learning [46,47].

With the focus of more attention and efforts, deep learning has burgeoned in recent years and has been applied broadly in industry. For instance, deep belief networks (DBNs) and stacks of restricted Boltzmann machines (RBMs) [3,48,49] have been applied in speech and image recognition [3,45,50] and natural language processing [51]. Proposed to better mimick animals' perceptions of objects [52], convolutional neural networks (CNN) have been widely applied in image recognition [53–55], image segmentation [56], video recognition [57,58], and natural language processing [59]. Recurrent neural networks (RNNs) are another class of ANNs that exhibit dynamic behavior, with artificial neurons that are associated with time steps [25,60,61].

4

RNNs have become the primary tool for handling sequential data [62], and have been applied in natural language processing [63] and handwriting recognition [64]. Later on, variants of AEs, including sparse AEs, stacked AEs (SAEs), and de-noising AEs, have also gained popularity in pre-training deep networks [49,65–67].

Although applications of deep learning have been primarily focused on image recognition, video and sound analyses, as well as natural language processing, it also opens doors in life sciences, which will be discussed in detail in the next sections.

Brief description of deep learning

Although the underlying assumptions and theories are different, the basic idea and processes for feature extraction in most deep NN (DNN) architectures are similar. In the forward pass, the network is activated by an input to the first layer, which then spreads the activation to the final layer along the weighted connections, and generates the prediction or reconstruction results. In the backward pass, the weights of connections are tuned by minimizing the difference between the predicted and the real data.

Basic concepts

Activation functions

Activation functions form the non-linear layers in all deep learning frameworks; and their combinations with other layers are used to simulate the non-linear transformation from the input to the output [62]. Therefore, better feature extraction can be achieved by selecting appropriate activation functions [7,68,69]. Here, we introduce several commonly-used activation functions, represented by g.

- Sigmoid function: $g(a) = \dfrac{1}{1+e^{-a}}$, where a is the input from the front layer. A sigmoid function transforms variables to values ranging from 0 to 1 and is commonly used to produce a Bernoulli distribution. For example:

$$\tilde{g} = \begin{cases} 0, & if\ g(a) \le 0.5 \\ 1, & if\ g(a) > 0.5 \end{cases},$$

- Hyperbolic tangent: $g(a) = \tanh(a)$ $\tanh a = \dfrac{\sinh a}{\cosh a} = \dfrac{e^a - e^{-a}}{e^a + e^{-a}}$. Here, the derivative of g is calculated as $g' = 1 - g^2$, making it easy to work with in BP algorithms.

- Softmax: $g(a) = \dfrac{e^{a_i}}{\sum\limits_j e^{a_j}}$. The softmax output, which a_n be considered as a probability

  distribution over the categories, is commonly used in the final layer.

- Rectified linear unit (ReLU): $g(a) = \max(0, a)$. This activation function and its variants show superior performance in many cases and are the most popular activation function in deep learning so far [68,70–72]. ReLU can also solve the gradient diffusion problem [73,74].

- Softplus: $g(a) = \log(1 + e^a)$. This is one of the variants of ReLU, representing a smooth approximation of ReLU (in this article, the log always represents the natural logarithm).

- Absolute value rectification: $g(a) = |a|$. This function is useful when the pooling layer takes the average value in CNNs [75], thus preventing otherwise the negative features and the positive features from diminishing.

- Maxout: $g_i(x) = \max\limits_i (b_i + w_i \cdot x)$. The weight matrix in this function is a three-dimensional array, where the third array corresponds to the connection of the neighboring layers [76].

Optimization objective

An optimization objective is often composed of a loss function and a regularization term. The loss function measures the discrepancy between the output of the network depend on model parameters (θ) $f(x|\theta)$ and the expected result y, *e.g.*, the true class labels in classification tasks, or the true level in prediction tasks. However, a good learning algorithm performs well not only on the training data, but also on the test data. A collection of strategies designed to reduce the test error is called regularization [62]. Some regularization terms apply penalties to parameters to prevent overly complex models. Here, we briefly introduce the commonly used loss function $L(f(x|\theta), y)$ and regularization term $\Omega(\theta)$. The optimization objective is usually defined as:

$$\tilde{L}(X, y, \theta) = L(f(x|\theta), y) + \alpha\, \Omega(\theta) \quad (1)$$

where α is a balance of these two components, and in practice, the loss function is usually calculated across randomly-sampled training samples rather than the data-generating distribution, since the latter is unknown.

Loss function

Most DNNs use cross entropy between the training data and the model distribution as the loss function. The most commonly used form of cross entropy is the negative conditional log-likelihood: $L(f(x|\theta),y) = -\log P(f = y|x,\theta)$. This is a collection of loss functions corresponding to the distribution of y given the value of input variable x. Here, we introduce several commonly used loss functions that follow this pattern:

Suppose y is continuous and has a Gaussian distribution over a given variable x. The loss function would be:

$$L(f(x|\theta), y) = -\log[\sqrt{\frac{1}{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}(y-f)^2)] = \frac{1}{2\sigma^2}(y-f)^2 + \frac{1}{2}\log(2\pi\sigma^2) \quad (2)$$

Which is equivalently described as the squared error. The squared error was the most commonly used loss function in the 1980s [62]. However, it often tends to penalize outliers excessively, leading to slower convergence rates [77].

If y follows the Bernoulli distribution, then the loss function will be:

$$L(f(x|\theta), y) = -y\log f(x|\theta) - (1-y)\log(1 - f(x|\theta)) \quad (3)$$

When y is discrete and has only two values, for instance, $y \in \{1,2,\cdots,k\}$, we can take the softmax value (see commonly-used activation functions) as the probability over the categories. Then the loss function will be:

$$L(f(x|\theta), y) = -\log(\frac{e^{a_y}}{\sum_j e^{a_j}}) = -a_y + \log(\sum_j e^{a_j}) \quad (4)$$

Regularization term

L^2 parameter regularization is the most common form of regularization term and contributes to the convexity of the optimization objective, leading to an easy solution for the minimum using the Hessian matrix [78,79]. L^2 parameter regularization can be defined as

$$\Omega(\theta) = \frac{1}{2}\|\omega\|^2 \quad (5)$$

where ω represents weights of connecting units in the network (the same as in the following context).

Compared to L^2 parameter regularization, L^1 parameter regularization results in a sparser solution of ω and tends to learn small groups of features. L^1 parameter regularization can be defined as

$$\Omega(\theta) = \| \omega \|_1 = \sum_i | \omega_i | \quad (6)$$

Frobenius parameter regularization is induced by the inner product and is block decomposable, therefore it is easier to compute [80,81]. Frobenius parameter regularization can be defined as

$$\Omega(\theta) = \sqrt{\sum_i \sum_j | \omega_{ij} |^2} = \sqrt{\sum_{i=1}^{rank(\omega)} \sigma_i^2} \quad (7)$$

where σ_i is the i-th largest singular value. Frobenius parameter regularization has a function similar to nuclear norm in terms of regularization.

Nuclear norm has been widely used as regularization in recent years [82–84]. Nuclear norm regularization measures the sum of the singular values of ω and can be defined as

$$\Omega(\theta) = \| \omega \|_* = \sum_{i=1}^{rank(\omega)} \sigma_i \quad (8)$$

Optimization methods

A learning task is transformed to an optimization problem, to achieve the minima of the objective function by selecting appropriate hyperparameters. The basic processes of different optimization methods are similar. First, the output $f = f(x | \theta_0)$ and the optimization objective $\widetilde{L}$ of the model are computed using the initial parameters θ_0. The network parameters θ are then tuned to decrease the objective function value from the final layer to the first layer [18]. This process is repeated until the proper model and a small fit error, *i.e.*, loss function value, are obtained (http://deeplearning.stanford.edu/wiki/index.php/UFLDL_Tutorial).

However, different optimization methods have different advantages and disadvantages on different architectures and loss functions [62,85]. Stochastic gradient descent (SGD) and its variants are the most-used methods, which update the parameters by a gap corresponding to the

Jacobian matrix. The computation time per update does not grow too much even with a large training set [86–88]. AdaGrad updates parameters according to the accumulation of squared gradients, which can converge rapidly when applied to convex functions, but performs worse in certain models [62]. RMSProp, an AdaGrad algorithm, has been an effective and popular method for parameter optimization. Another type of algorithm makes use of second order derivatives to improve optimization. For instance, limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) is one type of quasi-Newton method, which iteratively refines the approximation of the inverse of the Hessian matrix and avoids storing the matrix. BFGS is good at dealing with low dimensionality problems, particularly for convolutional models [85]. In addition, conjugate gradient combines conjugacy and gradient descent in the update direction decision for parameters, efficiently avoiding the calculation of the inverse Hessian [4,35,36], while contrastive divergence is usually used in RBM model [89–91]. With the help of a GPU [47], many algorithms can be accelerated significantly [85].

The proper architecture and objective function should be selected according to data considered. As a type of machine learning, deep learning can also encounter "overfitting," that is, low error on training data but high error on test data. In addition to the regularization terms, other methods for regularization are also important for reducing test error. Adding noise to the input or to the weights are efficient regularization strategies [41,92], as in the case of a denoising AE [93]. Stopping the optimization early by setting an iteration number is another commonly used strategy to prevent the network from overfitting [62]. Parameter sharing, just like in CNN, can also contribute to regularization [94]. Dropout can force units to independently evolve, and randomly remove portions of units in ANN on each iteration, and can therefore achieve better results with inexpensive computation [73,95,96].

Deep learning architectures

AEs

Different from ordinary ANNs, AEs extract features from unlabeled data and set target values to be equal to the inputs [4,49,97]. Given the input vector $\{x^{(1)}, x^{(2)}, x^{(3)}, \ldots\}, x^{(i)} \in R^n$, the AE tries to learn the model:

$$h_{W,b}(x) = g(Wx + b) \approx x \qquad (9)$$

where W and b are the parameters of the model, g is the activation function (same definition applied in the following context), and $h_{W,b}$ represents the hidden units. When the number of hidden units, which represents the dimension of features, is smaller than the input dimension, the AE performs a reduction of data dimensionality similar to principal component analysis [98]. Besides pattern recognition, an AE with a classifier in the final layer can perform classification tasks as well.

RBMs and DBNs

RBMs are generative graphical models that aim to learn the distribution of training data. Since we do not know which distribution the data obeys, we cannot directly compute model parameters using the maximum likelihood principle. Boltzmann machines (BMs) use an energy function to generate the probability distribution (see Equations 12 and 13 below), and then optimize parameters until the model learns the true distribution of the data. The original BMs have not been demonstrated to be useful for practical problems, while RBMs are commonly used in deep learning.

RBMs restrict the BMs to a bipartite graph, *i.e.*, there are no connections within visible units $v = x$ or hidden units h. This restriction ensures the conditional independency of hidden units and visible units [91], *i.e.*,

$$p(h\,|\,v) = \prod_i p(h_i\,|\,v)$$
$$p(v\,|\,h) = \prod_j p(v_j\,|\,h) \quad (10)$$

Furthermore, most RBMs rely on the assumption that all units in the network take only one of the two possible values 0 or 1, *i.e.*, $v_j, h_i \in (0,1)$. Provided with the activation function, the conditional distribution of hidden and visible units can be expressed in the following form:

$$p(h_i = 1\,|\,v) = g(W_i v + c_i)$$
$$p(v_j = 1\,|\,h) = g(W_j' h + b_j) \quad (11)$$

According to the Boltzmann distribution, probability distributions over hidden and visible vectors are defined as:

$$P(v,h) = \frac{1}{Z} e^{-E(v,h)} \quad (12)$$

10

where $Z = \sum e^{-E(v,h)}$ is the normalizing constant and $E(v,h) = -b'v - c'h - h'Wv$ is the energy function [99]. The conditional probability distribution can also be computed by integral, and the parameters can then be optimized by minimizing the Kullback-Leibler divergence.

Overall, given the network architectures and optimized parameters, the distribution of the visible units could be computed as:

$$p(v) = \sum_h p(v,h) = \sum_h \frac{e^{-E(v,h)}}{Z} \quad (13)$$

A DBN can be viewed as a stack of RBMs [6,24,100] or AEs [66,101]. Similar to RBMs, DBNs can learn the distribution of the samples, or learn to classify the inputs given class labels [3]. However, the $p(h)$ in the formula $p(v) = \sum_h p(v,h) = \sum_h p(h)p(v|h)$ is replaced by a better model after the weight of connections W is learned by an RBM [3,100].

In addition to feature extraction, RBMs can also learn distributions of unlabeled data as generative models, and classify labeled data as discriminative models (regard the hidden units as labels). Similar to AEs, RBMs can also pre-train parameters for a complex network.

Convolutional neural networks

Different from other deep learning structures, artificial neurons in convolutional neural networks (CNNs) extract features of small portions of input images, which are called receptive fields. This type of feature extraction was inspired by the visual mechanisms in living organisms, where cells in the visual cortex are sensitive to small regions of the visual field [52,102].

Besides the activation function, there are two particular types of layers in CNNs: the convolutional layer and the pooling layer (**Figure 2**). In the convolutional layer, the image is convolved by different convolutional filters via shifting the receptive fields step by step [87] (Figure 2A). The convolutional filters share the same parameters in every small portion of the image, largely reducing the number of hyperparameters in the model. A pooling layer, taking advantage of the "stationarity" property of images, takes the mean, the max, or other statistics of the features at various locations in the feature maps, thus reducing the variance and capturing essential features (http://deeplearning.net/tutorial/lenet.html) (Figure 2B).

Recurrent neural networks

11

Recurrent neural networks (RNNs) outperform other deep learning approaches in dealing with the sequential data. Based on the property of sequential data, parameters across different time steps of the RNN model are shared. Taking speech as an example: some vowels may last longer than other sounds; the difference makes absolute time steps meaningless and demands that the model parameters be the same among the time steps [62].

Beside the parameter sharing, RNNs are different from other multilayer networks by virtue of having a circuit, which represents hidden-to-hidden recurrence. A simple recurrent network corresponds to the following equation:

$$h_{(t)} = g(b + Uh_{(t-1)} + Wx_{(t)})$$
$$o_{(t)} = c + Vh_{(t)}$$

(14)

where t is the label for time, W and V represent the weights connecting hidden and input units, and hidden and output units, respectively, b and c are the offsets of the visible and hidden layers, respectively, g is the activation function, and U represents the weights connecting hidden units at time $t-1$ to hidden units at time t (**Figure 3**).

Similar to other deep learning architectures, RNNs can also be trained using the BP method. A variant of the BP method called back propagation through time (BPTT) is the standard optimization method for RNNs [25,103], and some alternative methods have also been proposed to speed up the optimization or to extend its capacity [63,104–107].

Applications in biomedicine

Owing to advances in high-throughput technologies, a deluge of biological and medical data has been obtained in recent decades, including data related to medical images, biological sequences, and protein structures. Some successful applications of deep learning in biomedical fields are reviewed in this section and a summary of applications is shown in **Table 1**.

Medical image classification and segmentation

Machine learning for medical images has long been a powerful tool in the diagnosis or assessment of diseases. Traditionally, discriminative features referring to medical image interpretation are manually designed for classification (detection of lesions or abnormalities) and segmentation of regions of interest (tissues and organs) in different medical applications. This requires the participation of physicians with expertise. Nonetheless, the complexity and

ambiguity of medical images, limited knowledge for medical image interpretation, and the requirement of large amounts of annotated data have hindered the wide use of machine learning in the medical image domain. Notably, deep learning methods have attained success in a variety of computer vision tasks such as object recognition, localization, and segmentation in natural images. These have soon brought about an active field of machine learning in medical image analysis.

Segmentation of tissues and organs is crucial for qualitative and quantitative assessment of medical images. Pereira et al. used data augmentation, small convolutional kernels, and a pre-processing stage to achieve accurate brain tumor segmentation [108]. Their CNN-based segmentation method won first place in the Brain Tumor Segmentation (BRATS) Challenge in 2013, and second place in 2015. Havaei et al. presented a fully automatic brain tumor segmentation method based on DNNs in magnetic resonance (MR) images with a two-phase training procedure [109], which obtained second place in the 2013 BRATS. Their methodology was tested on the publicly available datasets INbreast [110] and Digital Database for Screening Mammography (DDSM) [111], outperforming in terms of accuracy and efficiency several state-of-the-art methods when tested on DDSM. Additional medical applications employing a deep learning architecture have been demonstrated in segmenting the left ventricle of the heart from the MR data [112], the pancreas through computed tomography (CT) [113], tibial cartilage through magnetic resonance imaging (MRI) [114], the prostate through MRI [115], and the hippocampus through MR brain images [116,117]. The differentiation of tissues or organs in medical images has been termed semantic segmentation [118,119] in which each pixel of an image is assigned to a class or a label. The skeletal muscles, organs, and fat in CT images are well delineated through semantic segmentation based on a DNN architecture [120]. Similarly, the semantic segmentation of MR images also attained accurate segmentation results [121–123].

Detection of lesion and abnormality is the major issue in medical image analysis. Deep learning methods learn the representations directly instead of using hand-crafted features from training data. A classifier is then used to assign the representations to a probability that indicates whether or not the image contains lesions. In other words, the deep learning schemas classify each pixel to be a lesion point or not, which can be done in two ways: (1) classifying the mini patch around the pixel with a deep network, and (2) using a fully convolutional network to classify each pixel.

Debdoot et al. [124] applied a DNN to histologically characterize healthy skin and healing wounds to reduce clinical reporting variability. Two unsupervised pre-trained layers of denoising AEs (DAEs) were used to learn features in their hybrid architecture, and subsequently the whole network was learned using labelled tissues for characterization. Detection of cerebral microbleeds [125] and coronary artery calcification [126] also produced better results when using deep learning-based approaches. In addition, brain tumor progression prediction implemented with a deep learning architecture [127] has also shown a more robust tumor progression model in comparison with a high-precision manifold learning approach [128].

Detection of pathologies on stained histopathology images [129–131] exemplify the high precision of deep learning-based approaches. For breast cancer detection in histopathology images, Cruz-Roa et al. [132] established a deep learning model to precisely delineate the invasive ductal carcinoma (IDC) regions to distinguish the invasive tumor tissue and non-invasive or healthy tissue. Their 3-layer CNN architecture, composed of two cascading convolutional and pooling layers, a full-connected layer, and a logistic regression classifier for prediction, attained a better F-measure (71.8%) and higher balanced accuracy (BAC; 84.23%) in comparison with an approach using handcrafted image features and a machine learning classifier.

The mammogram is one of the most effective imaging modalities in early diagnosis and risk prediction of breast cancer. A deep learning model [133] trained on a large dataset of 45,000 images attained performance similar to that of certified screening radiologists in mammographic lesion detection. Kallenberg et al. [134] investigated the scoring of percentage mammographic density (PMD) and mammographic texture (MT) related to prediction of breast cancer risk. They employed a sparse AE to learn deep hierarchical features from unlabeled mammograms. Multinomial logistic regression or softmax regression was then used as a classifier in the supervised training. As a result, the performance of their approach was comparable with that of the subjective and expensive manual PMD and MT scorings.

Color fundus photography is an important diagnostic tool for ophthalmic diseases. Deep learning-based methods with fundus images have recently gained considerable interest as a key to developing automated diagnosis systems. A DNN architecture was proposed by Srivastava et al. [135] to distinguish optic disc (OD) from parapapillary atrophy (PPA). A DNN consisting of SAEs followed by a refined active shape model attained accurate OD segmentation. For image registration, deep learning in combination with a multi-scale Hessian matrix [136] was used to

detect vessel landmarks in the retinal image, whereas convolutional neural networks have also produced excellent results in the detection of hemorrhages [137] and exudates [138] in color fundus images. It is difficult to design an automatic screening system for retinal-based diseases such as age-related molecular degeneration, diabetic retinopathy, retinoblastoma, retinal detachment, and retinitis pigmentosa, because these diseases share similar characteristics. Through deep learning methods, Arunkumar et al. [139] successfully built a system to discriminate retina-based diseases only using fundus images. First, a DBN composed of a stack of RBMs was designed for feature extraction. Then a generalized regression neural network (GRNN) was employed to reduce dimensionality. Finally, a multi-class SVM was used for classification. Interestingly, Kaggle organized a competition on the staging of diabetic retinopathy from 35,126 training and 53,576 test color fundus images in 2015. Using convolutional neural networks, the top model outperformed other machine learning methods with a kappa score of 0.8496 (https://www.kaggle.com/c/diabetic-retinopathy-detection/leaderboard).

In addition to static images, time-series medical records such as signal maps from electro-encephalography and magnetoencephalography can also be analyzed using deep learning methods [140,141]. These deep learning schemas take coded features of signals [142,143] or raw signals [144] as input, and extract features from the data for anomaly classification or understanding emotions.

All the aforementioned applications illustrate that as a frontier of machine learning, deep learning has made substantial progress in medical image segmentation and classification. We expect that more clinical trials and systematic medical image analytic applications will emerge to help achieve better performance when applying deep learning in medicine.

Genomic sequencing and gene expression analysis

Deep learning also plays an important role in genomic sequencing and gene expression analyses. To infer the expression profiles of target genes based on approximately 1000 landmark genes from the NIH Integrated Network-based Cellular Signatures (LINCS) program, Chen et al. presented D-GEX, a deep learning method with dropout as regularization, which significantly outperformed linear regression (LR) in terms of prediction accuracy on both microarray and RNA-seq data [145]. By applying a multimodal DBN to model structural binding preferences and to predict binding sites of RNA-binding proteins (RBPs) using the primary sequence as well

as the secondary and tertiary structural profiles, Zhang et al. achieved an AUC of 0.98 for some proteins [146]. To predict binding sites of DNA- and RNA-binding proteins, Alipanahi et al. developed DeepBind, a CNN-based method, which surpassed other state-of-the-art methods, even when trained with *in vitro* data and tested with *in vivo* data [147]. Subsequently, Lanchantin et al. [148] and Zeng et al. [149] also applied CNN to predict transcription factor binding sites (TFBSs), and both studies demonstrated an improvement over the performance of DeepBind (AUC of 0.894). The input of these deep CNNs is encoded sequence characters obtained through protein binding microarrays or other assays, and the output is a real value indicating whether the sequence is a binding site or not. The deeper model can make more accurate classification by extracting higher-level features from the raw nucleotide sequences [148]. In addition, Kelley et al. presented Basset, an open source package to apply deep CNNs to learn the chromatin accessibility code, enabling annotation and interpretation of the noncoding genome [150]. Other applications include that of Li et al. [134] and Liu et al. [151,152], who proposed deep learning approaches for the identification of *cis*-regulatory regions and replication timing domains, respectively. In addition, Yoon and his collaborators employed RNNs to predict miRNA precursors and targets. As a result, they achieved 25% increase in F-measure compared to existing alternative methods [153,154].

Genetic variation can influence the transcription of DNA and the translation of mRNA [155]. Understanding the effects of sequence variants on pre-mRNA splicing facilitates not only whole genome annotation but also an understanding of genome function. To predict splice junction at the DNA level, Yoon and his collaborators developed a novel DBN-based method that was trained on the RBMs by boosting contrastive divergence with categorical gradients [156]. Their method not only achieved better accuracy and robustness but also discovered subtle non-canonical splicing patterns [156]. Furthermore, by exploiting RNNs to model and detect splice junctions from DNA sequences, the same authors also achieved a better performance than the previous DBN-based method [157].

Frey et al. formulated the assembly of a splicing code as a statistical inference problem [158], and proposed a Bayesian method to predict tissue-regulated splicing using RNA sequences and cellular context. Subsequently, they developed a DNN model with dropout to learn and predict alternative splicing (AS) [159]. This model took both the genomic features and tissue context as inputs, and predicted splicing patterns in individual tissues and differences in

splicing patterns across tissues. They showed that their method surpassed the previous Bayesian methods and other common machine learning algorithms, such as multinomial logistic regression (MLR) and SVMs, in terms of AS prediction. Furthermore, they built a computational model using a Bayesian deep learning algorithm to predict the effects of genetic variants on AS [160]. This model took DNA sequences alone as input without using disease annotations or population data, and then scored the effects that variants had on AS, providing valuable insights into the genetic determinants of spinal muscular atrophy, nonpolyposis colorectal cancer, and autism spectrum disorder.

To annotate the pathogenicity of genetic variants, Quang et al. developed a DNN algorithm named DANN, which outperforms logistic regression (LR) and SVMs, with the AUC metric increased by 14% over SVMs [161]. Zhou et al. proposed a CNN-based algorithmic framework, DeepSEA, to predict the functional effects of noncoding variants *de novo* from sequences [162]. DeepSEA directly learns a regulatory sequence code from large-scale chromatin-profiling data, and can then predict the chromatin effects of sequence alterations with single-nucleotide sensitivity, and further prioritize functional variants based on the predicted chromatin effect signals. Subsequently, DanQ, a novel hybrid framework that combines CNN and bi-directional long short-term memory (BLSTM) RNNs, was presented to predict non-coding function *de novo* from sequences alone [163]. DanQ achieved an AUC 50% higher than other models, including the aforementioned DeepSEA.

Prediction of protein structure

The 3D structure of proteins is determined by their comprising amino acid sequence [164]. However, the computational prediction of 3D protein structure from the 1D sequences remains challenging [165]. The correct 3D structure of a protein is crucial to its function, and improper structures could lead to a wide range of diseases [166–168]. Deep learning technologies have shown great capabilities in the area of protein structure prediction, which aims to predict the secondary structure or contact map of a protein.

Lyons et al. reported the first SAE for sequence-based prediction of backbone Cα angles and dihedrals [169]. Heffernan et al. also employed SAEs to predict secondary structure, local backbone angles, and solvent-accessible surface area (ASA) of proteins from amino acid sequences [170]; they achieved an accuracy of 82% for secondary structure prediction. Spencer

et al. proposed DNSS, an *ab initio* approach to predicting the secondary structure of proteins using deep learning network architectures [171]. DNSS was trained using a position-specific scoring matrix of the protein sequence and Atchley's factors of residues, and was optimized to accelerate the computation using the GPU and compute unified device architecture (CUDA). Baldi and his colleagues successfully applied various RNN-based algorithms to predict protein secondary structure [172–174] and protein contact map [175–177], with accuracies of 84% and 30%, respectively. Sønderby et al. used a bidirectional RNN (BRNN) with long short-term memory cells to improve the prediction of secondary structure, with better accuracy (0.671) than that using state of the art (0.664) [178]. Compared with SAEs, DBNs, and RNNs, CNNs were seldom used for protein structure prediction until recently. Li et al. developed Malphite, a CNN and ensemble learning-based method for predicting protein secondary structures, which achieved an accuracy of 82.6% for a dataset containing 3000 proteins [179]. Additionally, Lin et al. proposed MUST-CNN, a multilayer shift-and-stitch convolutional neural network architecture to predict protein secondary structure from primary amino acid sequences [180]. Besides classical deep learning architectures, some other architectures were also employed to predict protein secondary structure. For example, Lena et al. introduced a deep spatio-temporal learning architecture, achieved an accuracy roughly 10% higher than other methods [181], and Zhou et al. presented a deep supervised and convolutional generative stochastic network, achieving an accuracy of 66.4% [182].

In addition to the secondary structure prediction, deep learning was also employed in protein region prediction [183,184]. For instance, sequenced-based predictor of protein disorder using boosted ensembles of deep networks (DNdisorder), a deep neural network with multi-layers of RBMs [184], achieved an average balanced accuracy of 0.82 and an AUC of 0.90. Incorporated with predicted secondary structure and predicted ASA, a weighted deep convolutional neural fields (DeepCNF) was proposed to predict protein order/disorder regions, obtains an AUC of 0.898 on the Critical As-sessment of Techniques for Protein Structure Prediction (CASP10) dataset [183]. All of these methods surpassed other state-of-the-art predictors in accuracy while still maintaining an extremely high computing speed. Recently, RaptorX-Property, a web server employing DeepCNF, was also presented to predict protein structure properties, including secondary structure, solvent accessibility, and disorder regions [185]. RaptorX-Property can be easily used and offer good performance (an AUC of 0.89 on its test data).

Conclusion and perspective

Deep learning is moving toward its original goal: artificial intelligence. The state-of-the-art feature extraction capacity of deep learning enables its application in a wide range of fields. Many deep learning frameworks are open source, including commonly-used frameworks like Torch, Caffe, Theano, MXNet, DMTK, and TensorFlow. Some of them are designed as high-level wrappers for easy use, such as Keras, Lasagne, and Blocks. The applications of deep learning algorithms is further facilitated by the freely available sources. **Figure 4** summarizes commonly-used frameworks in Github (https://github.com/) where the number of stars reflects the popularity of the frameworks.

Breakthroughs in technologies, particularly next-generation sequencing, are producing a large quantity of genomic data. Efficient interpretation of these data has been attracting much attention in recent years. In this scenario, uncovering the relationship between genomic variants and diseases, and illustrating the regulatory process of genes in cells have been important research areas. In this review, we introduced the way deep learning gets involved in these areas using examples. With deep architecture, these models can simulate more complex transformations and discover hierarchical data representations. On the other hand, almost all of these models can be trained in parallel on GPUs for fast processing. Furthermore, deep learning can extract data-driven features and deal with high-dimensional data, while machine learning usually depends on hand-crafted features and is suitable only to low-dimensional data. Thus, deep learning is becoming more and more popular in genomic sequence analysis.

Deep learning is represented by a group of technologies (introduced in brief description of deep learning), and has been widely used in biomedical data (introduced in applications in biomedicine). SAEs and RBMs can extract patterns from unlabeled data [186] as well as labeled data when stacked with a classifier [156]. They can also deal with dynamic data [187]. CNNs are most commonly used in the biomedical image analysis domain due to their outstanding capacity in analyzing spatial information. Although relatively few CNNs are used in sequencing data, CNNs have great potential in omics analysis [147] and biomedical signals [142]. On the other hand, RNN-based architectures are tailored for sequential data, and are most often used for sequencing data [154,157] and in dynamic biomedical signals [144], but less frequently in static biomedical images. Currently, more and more attention is being paid to the usage of deep

learning in biomedical information, and new applications of each schema may be discovered in the near future.

Despite the notable advantages of deep learning, challenges in applying deep learning to the biomedical domain still remain. Take biomedical image analysis for instance: we use fundus images to exemplify how deep learning works to define the level of diabetic retinopathy, and to detect lesion areas in different ways. Besides high accuracy and speed, the intelligent use of receptive fields also endows deep learning with overwhelming superiority in terms of image recognition. Furthermore, the development of end-to-end classification methods based on deep learning sheds new light on classifying pixels as lesioned or not. However, the usage of deep learning in medical images is still challenging. For model training, we need large amounts of data with labels, sometimes with labels in terms of pixel classification. Manually labeling these medical images is laborious and requires professional experts. On the other hand, medical images are highly associated with privacy, so collecting and protecting the data is demanding. Furthermore, biomedical data are usually imbalanced because the quantity of data from normal classes is much larger than that from other classes.

In addition to the balancing challenges, the large amount of data required, and the labeling for biomedical data, deep learning also requires technological improvements. Unlike other images, subtle changes in medical images may indicate disease. Therefore, analyzing these images requires high-resolution inputs, high training speed, and a large memory. Additionally, it is difficult to find a uniform assessment metric for biomedical data classification or prediction. Unlike other projects, we can tolerate false positives to some extent, and reject few or no false negatives in disease diagnosis. With different data, it is necessary to assess the model carefully and to tune the model according to characteristics of the data. Fortunately, the deeper networks with inception modules are accelerated [188,189] and provide higher accuracy in biomedical image analysis [190]. On the other hand, crowdsourcing approaches have begun to pave the way in collecting annotations [191,192], which may be an important tool in the next few years. These bidirectional drivers would promote the applications of deep learning in biomedical informatics.

As a long-term goal, precision medicine research demands active learning from all biological, biomedical, as well as health data. Together with medical devices and instruments, wearable sensors and smart phones are providing unprecedented amounts of health data. Deep learning is a promising interpreter of these data, serving in disease prediction, prevention,

diagnosis, prognosis, and therapy. We expect that more deep learning applications will be available in epidemic prediction, disease prevention, and clinical decision-making.

Competing interests

The authors have declared no competing interests.

Acknowledgments

References

[1]   Yu D, Deng L. Deep learning and its applications to signal and information processing. IEEE Signal Process Mag 2011;28:145–54.

[2]   Fukushima K. Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol Cybern 1980;36:193–202.

[3]   Hinton GE, Osindero S, Teh YW. A fast learning algorithm for deep belief nets. Neural Comput 2006;18:1527–54.

[4]   Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. Science 2006;313:504–7.

[5]   Cios KJ, Mamitsuka H, Nagashima T, Tadeusiewicz R. Computational intelligence in solving bioinformatics problems. Artif Intell Med 2005;35:1–8.

[6]   Längkvist M, Karlsson L, Loutfi A. A review of unsupervised feature learning and deep learning for time-series modeling. Pattern Recognit Lett 2014;42:11–24.

[7]   Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Adv Neural Inform Process Syst 2012;60:1097–105.

[8]   Asgari E, Mofrad MRK. ProtVec: a continuous distributed representation of biological sequences. arXiv1503.05140v1.

[9]   Hubel DH, Wiesel TN. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J Physiol 1962;160:106–54.

[10]   Hubel DH, Wiesel TN. Receptive fields of single neurones in the cat's striate cortex. J Physiol 1959;148:574–91.

[11]   Weng J, Ahuja N, Huang TS. Cresceptron: a self-organizing neural network which grows adaptively. Proc Int Jt Conf Neural Netw 1992;1:576–81.

[12]   Weng JJ, Ahuja N, Huang TS. Learning recognition and segmentation of 3-D objects from 2-D images. Proc IEEE Int Conf Comput Vis 1993:121–8.

[13]   Weng J, Ahuja N, Huang TS. Learning recognition and segmentation using the cresceptron. Int J Comput Vis 1997;25:109–43.

[14]   Riesenhuber M, Poggio T. Hierarchical models of object recognition in cortex. Nat Neurosci 1999;2:1019–25.

[15]   Joseph, Roger D. Contributions to perceptron theory. In: Ph.D. thesis, Cornell University; 1961.

[16]   Viglione SS. Applications of pattern recognition technology. In: Mendel JM, Fu KS, editors. Mathematics in science and engineering. Amsterdam: Elsevier B. V.; 1970, p.115–62.

[17]   Newell A, Papert MM. Perceptrons. An introduction to computational geometry. Science 1969;165:780–2.

[18]   Werbos P. Beyond regression: new tools for prediction and analysis in the behavioral sciences. In: Ph.D. dissertation, Harvard University; 1974,29:65–78.

[19]   Werbos P. Applications of advances in nonlinear sensitivity analysis. In: Drenick RF, Kozin F, editors. System modeling and optimization. Berlin: Springer Berlin Heidelberg; 1982, p. 762–70.

[20]   Werbos P. Backwards differentiation in ad and neural nets: past links and new opportunities. In: Bücker M, Corliss G, Naumann U, Hovland P, Norris B, editors. Automatic differentiation: applications, theory, and implementations. Berlin: Springer Berlin Heidelberg; 2006, p. 15–34.

[21]   LeCun Y. Une procédure d'apprentissage pour réseau à seuil asymétrique. Proc Cogn 1985:599–604.

[22]   LeCun Y. A theoretical framework for back-propagation. Proc 1988 Connect Model Summer Sch 1988:21–8.

[23]   Lang KJ, Waibel AH, Hinton GE. A time-delay neural network architecture for isolated word recognition. Neural Netw 1990;3:23–43.

[24]   Schmidhuber J. Deep learning in neural networks: an overview. Neural Netw 2015;61:85–117.

[25]   Rumelhart DE, McClelland JL, the PDP Research Group. Parallel distributed processing: explorations in the microstructure of cognition. Cambridge: MIT Press; 1986, p. 318–62.

[26]   West AHL, Saad D. Adaptive back-propagation in on-line learning of multilayer networks. NIPS'95 Proc 8th Int Conf Neural Inform Process Syst 1995:323–9.

[27]   Battiti R. Accelerated backpropagation learning: two optimization methods. Complex Syst 1989;3:331–42.

[28]   Almeida LB. Artificial neural networks. Piscataway: IEEE Press; 1990.

[29]   Marquardt DW. An algorithm for least-squares estimation of nonlinear parameters. J Soc Ind Appl Math 1963;11:431–41.

[30]   Gauss CF. Theoria motus corporum coelestium in sectionibus conicis solem ambientium. Cambridge: Cambridge University Press; 1809.

[31]   Broyden CG. A class of methods for solving nonlinear simultaneous equations. Math Comput 1965;19:577–93.

[32]   Fletcher R, Powell MJD. A rapidly convergent descent method for minimization. Comput J 1963;6:163–8.

[33]   Goldfarb D. A family of variable-metric methods derived by variational means. Math Comput 1970;24:23–26.

[34]   Shanno DF. Conditioning of quasi-Newton methods for function minimization. Math Comput 1970;24:647–656.

[35]   Møller M. Exact calculation of the product of the hessian matrix of feed-forward network error functions and a vector in 0 (n) time. Daimi Rep 1993:14.

[36]   Hestenes MR, Stiefel E. Methods of conjugate gradients for solving linear systems. J Res Nat Bur Stand 1952;49:409–36.

[37]   Cortes C, Vapnik V. Support-vector networks. Mach Learn 1995;20:273–97.

[38]   Ho TK. Random decision forests. Proc 3rd Int Conf Doc Anal Recognit 1995;1:278–82.

[39] Ho TK. The random subspace method for constructing decision forest. IEEE Trans Pattern Anal Mach Intell 1998;20:832–44.

[40] Altman NS. An introduction to kernel and nearest-neighbor nonparametric regression. Am Stat 1992;46:175–85.

[41] Graves A. Practical variational inference for neural networks. In: Shawe-Taylor J, Zemel RS, Bartlett PL, Pereira F, Weinberger KQ, editors. Advances in neural information processing systems. New York: Curran Associates Inc.; 2011, p. 2348–56.

[42] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Netw Learn Syst 1994;5:157–66.

[43] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44.

[44] Ciresan DC, Meier U, Masci J, Gambardella LM, Schmidhuber J. Flexible, high performance convolutional neural networks for image classification. IJCAI'11 Proc 22^{ed} Int Joint Conf Artif Intell 2011;22:1237–42.

[45] Hinton G, Deng L, Yu D, Dahl GE, Mohamed A, Jaitly N, et al. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. Signal Process Mag IEEE 2012;29:82–97.

[46] Cireşan DC, Meier U, Gambardella LM, Schmidhuber J. Deep, big, simple neural nets for handwritten digit recognition. Neural Comput 2010;22:3207–20.

[47] Raina R, Madhavan A, Ng AY. Large-scale deep unsupervised learning using graphics processors. ICML'09 Proc 26th Ann Int Conf Mach Learn 2009:873–80.

[48] Hinton GE. Boltzmann machine. Scholarpedia 2007;2:1668.

[49] Bengio Y. Learning deep architectures for AI. Delft: Now Pulishers Inc.; 2009,p.1–127.

[50] Sutskever I, Hinton GE. Learning multilevel distributed representations for high-dimensional sequences. J Mach Learn Res 2007;2:548–55.

[51] Sarikaya R, Hinton GE, Deoras A. Application of deep belief networks for natural language understanding. IEEE/ACM Trans Audio Speech Lang Process 2014;22:778–84.

[52] Matsugu M, Mori K, Mitari Y, Kaneda Y. Subject independent facial expression recognition with robust face detection using a convolutional neural network. Neural Netw 2003;16:555–9.

[53] Sermanet P, LeCun Y. Traffic sign recognition with multi-scale convolutional networks. Neural Netw 2011;42:3809−13.

[54] Lawrence S, Giles CL, Tsoi AC, Back AD. Face recognition: a convolutional neural-network approach. IEEE Trans Neural Netw Learn Syst 1997;8:98–113.

[55] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit. 2015:1−9.

[56] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2015;79:3431−40.

[57] Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Li FF. Large-scale video classification with convolutional neural networks. Proc IEEE Conf Comput Vis Pattern Recognit 2014:1725–32.

[58] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ, editors. Advances in neural information processing systems. New York: Curran Associates Inc.; 2014, p. 568–76.

[59] Collobert R, Weston J. A unified architecture for natural language processing: deep neural networks with multitask learning. ACM Proc Int Conf Mach Learn; 2008:160–7.

[60] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9:1735–80.

[61] Graves A. Supervised sequence labelling with recurrent neural networks. Berlin: Springer-Verlag Berlin Heidelberg; 2012.

[62]   Goodfellow I, Bengio Y, Courville A. Modern practical deep networks. In: Goodfellow I, Bengio Y, Courville A, editors. Deep learning. Cambridge: MIT Press; 2015, p. 162−481.

[63]   Gers FA, Schmidhuber J. LSTM recurrent networks learn simple context-free and context-sensitive languages. IEEE Trans Neural Netw 2001;12:1333–40.

[64]   Graves A, Schmidhuber J. Offline handwriting recognition with multidimensional recurrent neural networks. In: Koller D, Schuurmans D, Bengio Y, Bottou L, editors. Advances in neural information processing systems. New York: Curran Associates Inc.; 2009, p. 545–52.

[65]   Ballard DH. Modular learning in neural networks. Proc Conf AAAI Artif Intell 1987:279–84.

[66]   Schölkopf B, Platt J, Hofmann T. Greedy layer-wise training of deep networks. Adv Neural Inf Process Syst 2007:153–60.

[67]   Schölkopf B, Platt J, Hofmann T. Efficient sparse coding algorithms. Adv Neural Inf Process Syst 2007:801–8.

[68]   Bengio Y. Practical recommendations for gradient-based training of deep architectures. Lect Notes Comput Sci 2012;7700:437–78.

[69]   Singh RG, Kishore N. The impact of transformation function on the classification ability of complex valued extreme learning machines. Int Conf Control Comput Commun Mater 2013: 1–5.

[70]   Toth L. Phone recognition with deep sparse rectifier neural networks. Proc IEEE Int Conf Acoust Speech Signal Process 2013:6985–9.

[71]   Maas AL, Hannun AY, Ng AY. Rectifier nonlinearities improve neural network acoustic models. Proc 30th Int Conf Mach Learn 2013:30.

[72]   Nair V, Hinton GE. Rectified linear units improve restricted boltzmann machines. ICML'10 Proc 27th Int Conf Mach Learn 2010:807–14.

[73]   Lai M. Deep learning for medical image segmentation. arXiv150502000.

[74]   Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. J Mach Learn Res 2011; 15:315–23.

[75]   Jarrett K, Kavukcuoglu K, Ranzato M, LeCun Y. What is the best multi-stage architecture for object recognition? Proc IEEE Int Conf Comput Vis 2009:2146–53.

[76]   Goodfellow IJ, Warde-Farley D, Mirza M, Courville A. Maxout Networks. arXiv13024389.

[77]   Rosasco L, De Vito E, Caponnetto A, Piana M, Verri A. Are loss functions all the same? Neural Comput 2004;16:1063–76.

[78]   Binmore KG, Davies J. Calculus: concepts and methods. Cambridge: Cambridge University Press; 2002.

[79]   Boyd S, Vandenberghe L. Convex optimization. Cambridge: Cambridge University Press; 2004.

[80]   Huang J, Dong M, Li S. A new method of regularization parameter estimation for source localization. IEEE CIE Int Conf 2011;2:1804–8.

[81]   Yu Y, Schuurmans D. Rank / norm regularization with closed-form solutions : application to subspace clustering. Assoc Uncertain Artif Intell 2002:1–5.

[82]   Abernethy J, Bach F, Evgeniou T, Vert JP. A new approach to collaborative filtering: operator estimation with spectral regularization. J Mach Learn Res 2009;10:803–26.

[83]   Argyriou A, Evgeniou T, Pontil M. Convex multi-task feature learning. Mach Learn 2008;73:243–72.

[84]   Obozinski G, Taskar B, Jordan MI. Joint covariate selection and joint subspace selection for multiple classification problems. Stat Comput 2010;20:231–52.

[85]   Gauriau R, Cuingnet R, Lesage D, Bloch I. Multi-organ localization with cascaded global-to-local regression and shape prior. Med Image Anal 2015;23:70–83.

[86]   Bottou L. Stochastic gradient learning in neural networks. Proc Neuro Nımes 1991;91.

[87]   Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE 1998;86:2278–324.

[88]   Zinkevich M, Weimer M, Li L, Smola AJ. Parallelized stochastic gradient descent. In: Lafferty JD, Williams CKI, Shawe-Taylor J, Zemel RS, Culotta A, editors. Advances in Neural Information Processing Systems, New York: Curran Associates Inc.; 2010, p. 2595–603.

[89]   Hinton GE. Products of experts. ICANN 1999:1–6.

[90]   Hinton GE. Training products of experts by contrastive divergence. Neural Comput 2002:1771−800.

[91]   Carreira-Perpinan MA, Hinton GE. On contrastive divergence learning. Proc Artif Intell Stat 2005:1−17.

[92]   Jim KC, Giles CL, Horne BG. An analysis of noise in recurrent neural networks: convergence and generalization. IEEE Trans Neural Netw 1996;7:1424–38.

[93]   Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 2010;11:3371–408.

[94]   Lasserre JA, Bishop CM, Minka TP. Principled hybrids of generative and discriminative models. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2006;1:87–94.

[95]   Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. arXiv12070580.

[96]   Srivastava N, Hinton GE, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout : a simple way to prevent neural networks from overfitting. J Mach Learn Res 2014;15:1929–58.

[97]   Aurelio Ranzato M, Poultney C, Chopra S, Cun YL. Efficient learning of sparse representations with an energy-based model. In: Schölkopf B, Platt JC, Hoffman T, editors. Advances in neural information processing systems. New York: Curran Associates Inc.; 2007, p. 1137–44.

[98]   Bourlard H, Kamp Y. Auto-association by multilayer perceptrons and singular value decomposition. Biol Cybern 1988;59:291–4.

[99]   Hinton G. A practical guide to training restricted boltzmann machines. In: Montavon G, Orr G, Müller KR, editors. Neural networks: tricks of the Trade. Berlin: Springer Berlin Heidelberg; 2012, p. 599–619.

[100]   Hinton GE. Deep belief networks. Oxford: Oxford Univ Press; 2009, p. 5947.

[101]   Erhan D, Bengio Y, Courville A, Manzagol PA, Vincent P, Bengio S. Why does unsupervised pre-training help deep learning? J Mach Learn Res 2010;11:625–60.

[102]   Ciresan D, Meier U, Schmidhuber J. Multi-column deep neural networks for image classification. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2012:3642–9.

[103]   Werbos P. Generalization of backpropagation with application to a recurrent gas market model. Neural Netw 1988;1:339–56.

[104]   Pearlmutter B. Learning state space trajectories in recurrent neural networks. Neural Comput 2011;1:263–9.

[105]   Hochreiter S, Bengio Y, Frasconi P, Schmidhuber J. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In: Kolen JF, Kremer SC, editors. A field guide to dynamical recurrent neural networks. Wiley-IEEE press; 2001, p. 237–43.

[106]   Syed O. Applying genetic algorithms to recurrent neural networks for learning network parameters and architecture. Cleveland: Case Western Reserve University Press; 1995.

[107]   Gomez F, Schmidhuber J, Miikkulainen R. Accelerated neural evolution through cooperatively coevolved synapses. J Mach Learn Res 2008;9:937–65.

[108]   Pereira S, Pinto A, Alves V, Silva CA. Brain Tumor segmentation using convolutional neural networks in MRI images. IEEE Trans Med Imaging 2016;35:1240–51.

[109] Havaei M, Davy A, Warde-Farley D, Biard A, Courville A, Bengio Y, et al. Brain tumor segmentation with deep neural networks. Med Image Anal 2017;35:18–31.

[110] Moreira IC, Amaral I, Domingues I, Cardoso A, Cardoso MJ, Cardoso JS. INbreast: Toward a full-field digital mammographic database. Acad Radiol 2012;19:236–48.

[111] Health M, Bowyer K, Kopans D, Moore R, Kegelmeyer WP, Sallam M, et al. The digital database for screening mammography. In: Yaffe MJ, editor. Detection and characterization of mammographic masses by artificial neural network. Berlin: Springer Netherlands; 2001, p. 457–60.

[112] Ngo TA, Lu Z, Carneiro G. Combining deep learning and level set for the automated segmentation of the left ventricle of the heart from cardiac cine magnetic resonance. Med Image Anal 2017;35:159–71.

[113] Roth HR, Farag A, Lu L, Turkbey EB, Summers RM. Deep convolutional networks for pancreas segmentation in CT imaging. ArXiv1504.03967.

[114] Prasoon A, Petersen K, Igel C, Lauze F, Dam E, Nielsen M. Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network. Med Image Comput Comput Assist Interv 2013;8150:246–53.

[115] Liao S, Gao Y, Oto A, Shen D. Representation learning: A unified deep learning framework for automatic prostate MR segmentation. Med Image Comput Comput Assist Interv 2013;16:254–61.

[116] Guo YR, Wu GR, Commander LA, Szary S, Jewells V, Lin WL, et al. Segmenting hippocampus from infant brains by sparse patch matching with deep-learned features. Med Image Comput Comput Assist Interv 2014;8674:308–15.

[117] Kim M, Wu G, Shen D. Unsupervised deep learning for hippocampus segmentation in 7.0 tesla MR images. In: Wu G, Zhang D, Shen D, Yan P, Suzuki K, Wang F, editors. Proceedings of the 4th international workshop on machine learning in medical imaging. New York: Springer-Verlag New York Inc.; 2013:1–8.

[118] Schlegl T, Waldstein SM, Vogl WD, Schmidt-Erfurth U, Langs G. Predicting semantic descriptions from medical images with convolutional neural networks. Basel: Springer International Publishing AG; 2015, p. 437–48.

[119] Xu Y, Li Y, Liu M, Wang Y, Fan Y, Lai M, et al. Gland instance segmentation by deep multichannel neural networks. arXiv160704889.

[120] Lerouge J, Herault R, Chatelain C, Jardin F, Modzelewski R. IODA: an input/output deep architecture for image labeling. Pattern Recognition 2015;48:2847–58.

[121] Moeskops P, Viergever MA, Mendrik AM, de Vries LS, Benders MJNL, Isgum I. Automatic segmentation of MR brain images with a convolutional neural network. IEEE Trans Med Imaging 2016;35:1252–61.

[122] Shin HC, Orton MR, Collins DJ, Doran SJ, Leach MO. Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4D patient data. IEEE Trans Pattern Anal Mach Intell 2013;35:1930–43.

[123] Roth HR, Lee CT, Shin HCC, Seff A, Kim L, Yao J, et al. Anatomy-specific classification of medical images using deep convolutional nets. Proc IEEE Int Symp Biomed Imaging 2015: 101–4.

[124] Sheet D, Karri SPK, Katouzian A, Navab N, Ray AK, Chatterjee J. Deep learning of tissue specific speckle representations in optical coherence tomography and deeper exploration for in situ histology. Proc IEEE Int Symp Biomed Imaging 2015:777–80.

[125] Dou Q, Chen H, Yu L, Zhao L, Qin J, Wang D, et al. Automatic detection of cerebral microbleeds from MR images via 3D convolutional neural networks. IEEE Trans Med Imaging 2016;35:1182–95.

[126] Wolterink JM, Leiner T, de Vos BD, van Hamersvelt RW, Viergever MA, Išgum I. Automatic coronary artery calcium scoring in cardiac CT angiography using paired convolutional neural networks. Med Image Anal 2016;34:123–36.

[127] Zhou DQ, Tran L, Wang JH, Li J. A comparative study of two prediction models for brain tumor progression. Image Process Algorithms Syst 2015;9399.

[128] Tran L, Banerjee D, Wang J, Kumar AJ, Mckenzie F, Li Y, et al. High-dimensional MRI data analysis using a large-scale manifold learning approach. Mach Vis Appl 2013;24:995–1014.

[129] Sirinukunwattana K, Raza SEA, Tsang YW, Snead DRJ, Cree IA, Rajpoot NM. Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. IEEE Trans Med Imaging 2016;35:1196–206.

[130] Xu Y, Zhu JY, Chang E, Tu Z. Multiple clustered instance learning for histopathology cancer image classification, segmentation and clustering. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2012:964–71.

[131] Cireşan DC, Giusti A, Gambardella LM, Schmidhuber J. Mitosis detection in breast cancer histology images with deep neural networks. Med Image Comput Comput Assist Interv 2013: 411–8.

[132] Cruz-Roa A, Basavanhally A, Gonzalez F, Gilmore H, Feldman M, Ganesan S, et al. Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks. Med Imaging 2014 Digit Pathol 2014;9041.

[133] Kooi T, Litjens G, van Ginneken B, Gubern-Mérida A, Sánchez CI, Mann R, et al. Large scale deep learning for computer aided detection of mammographic lesions. Med Image Anal 2017;35:303–12.

[134] Kallenberg M, Petersen K, Nielsen M, Ng AY, Diao PF, Igel C, et al. Unsupervised deep learning applied to breast density segmentation and mammographic risk scoring. IEEE Trans Med Imaging 2016;35:1322–31.

[135] Srivastava R, Cheng J, Wong DWK, Liu J. Using deep learning for robustness to parapapillary atrophy in optic disc segmentation. 2015 IEEE 12th Int Symp Biomed Imaging 2015:768–71.

[136] Fang T, Su R, Xie L, Gu Q, Li Q, Liang P, et al. Retinal vessel landmark detection using deep learning and hessian matrix. Proc Int Symp Image Signal Process Anal 2015:387–92.

[137] Van Grinsven MJJP, van Ginneken B, Hoyng CB, Theelen T, Sanchez CI. Fast convolutional neural network training using selective data sampling: application to hemorrhage detection in color fundus images. IEEE Trans Med Imaging 2016;35:1273–84.

[138] Prentašić P, Lončarić S. Detection of exudates in fundus photographs using convolutional neural networks. Proc Int Symp Image Signal Process Anal 2015:188–92.

[139] Arunkumar R, Karthigaikumar P. Multi-retinal disease classification by reduced deep learning features. Neural Comput Appl 2015:1–6.

[140] Mirowski PW, LeCun Y, Madhavan D, Kuzniecky R. Comparing SVM and convolutional networks for epileptic seizure prediction from intracranial EEG. IEEE Int Workshop Mach Learn Signal Process 2008:244–9.

[141] Mirowski PW, Madhavan D, Lecun Y. Time-delay neural networks and independent component analysis for Eeg-Based prediction of epileptic seizures propagation. Proc Conf AAAI Artif Intell 2007:1892–3.

[142] Mirowski P, Madhavan D, LeCun Y, Kuzniecky R. Classification of patterns of EEG synchronization for seizure prediction. Clin Neurophysiol 2009; 120:1927–40.

[143] Davidson PR, Jones RD, Peiris MTR. EEG-based lapse detection with high temporal resolution. IEEE Trans Biomed Eng 2007;54:832–9.

[144] Petrosian A, Prokhorov D, Homan R, Dasheiff R, Wunsch D. Recurrent neural network based prediction of epileptic seizures in intra- and extracranial EEG. Neurocomputing 2000;30:201–18.

[145] Chen Y, Li Y, Narayan R, Subramanian A, Xie X. Gene expression inference with deep learning. Bioinfarmatics 2016;32:1832–9.

[146] Zhang S, Zhou J, Hu H, Gong H, Chen L, Cheng C, et al. A deep learning framework for modeling structural features of RNA-binding protein targets. Nucleic Acids Res 2016:44:e32.

[147] Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. Nat Biotechnol 2015;33:1–9.

[148] Lanchantin J, Singh R, Lin Z, Qi Y. Deep Motif: visualizing genomic sequence classifications. arXiv160501133.

[149] Zeng H, Edwards MD, Liu G, Gifford DK. Convolutional neural network architectures for predicting DNA-protein binding. Bioinformatics 2016;32:i121–7.

[150] Kelley DR, Snoek J, Rinn JL. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. Genome Res 2016;26:990–9.

[151] Liu F, Ren C, Li H, Zhou P, Bo X, Shu W. *De novo* identification of replication-timing domains in the human genome by deep learning. Bioinformatics 2015;32:641–9.

[152] Liu F, Li H, Ren C, Bo X, Shu W. PEDLA: predicting enhancers with a deep learning-based algorithmic framework. Sci Seq 2016;6:28517.

[153] Park S, Min S, Choi H, Yoon S. deepMiRGene: deep neural network based precursor microrna prediction. arXiv1605.00017.

[154] Lee B, Baek J, Park S, Yoon S. deepTarget: end-to-end learning framework for microRNA target prediction using deep recurrent neural networks. arXiv1603.09123.

[155] Guigo R, Valcarcel J. Prescribing splicing. Science 2015;347:124–5.

[156] Lee T, Yoon S. Boosted categorical restricted boltzmann machine for computational prediction of splice junctions. Proc Int Conf Mach Learn 2015;37.

[157] Lee B, Lee T, Na B, Yoon S. DNA-level splice junction prediction using deep recurrent neural networks. arXiv1512.05135.

[158] Xiong HY, Barash Y, Frey BJ. Bayesian prediction of tissue-regulated splicing using RNA sequence and cellular context. Bioinformatics 2011;27:2554–62.

[159] Leung MKK, Xiong HY, Lee LJ, Frey BJ. Deep learning of the tissue-regulated splicing code. Bioinformatics 2014;30:i121–9.

[160] Xiong HY, Alipanahi B, Lee LJ, Bretschneider H, Merico D, Yuen RKC, et al. The human splicing code reveals new insights into the genetic determinants of disease. Science 2014;347:1254806 .

[161] Quang D, Chen Y, Xie X. DANN: A deep learning approach for annotating the pathogenicity of genetic variants. Bioinformatics 2015;31:761–3.

[162] Zhou J, Troyanskaya OG. Predicting effects of noncoding variants with deep learning–based sequence model. Nat Methods 2015;12:931–4.

[163] Quang D, Xie X. DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. Nucleic Acids Res 2016;44:11.

[164] Anfinsen CB. The formation and stabilization of protein structure. Biochem J 1972;128:737–49.

[165] Gibson KD, Scheraga HA. Minimization of polypeptide energy. I. Preliminary structures of bovine pancreatic ribonuclease S-peptide. Proc Natl Acad Sci U S A 1967;58:420–7.

[166] Hammarstrom P, Wiseman RL, Powers ET, Kelly JW. Prevention of transthyretin amyloid disease by changing protein misfolding energetics. Science 2003;299:713–6.

[167] Chiti F, Dobson CM. Protein misfolding, functional amyloid, and human disease. Annu Rev Biochem 2006;75:333–66.

[168] Selkoe DJ. Folding proteins in fatal ways. Nature 2003;426:900–4.

[169] Lyons J, Dehzangi A, Heffernan R, Sharma A, Paliwal K, Sattar A, et al. Predicting backbone Cα angles and dihedrals from protein sequences by stacked sparse auto-encoder deep neural network. J Comput Chem 2014;35:2040–6.

[170] Heffernan R, Paliwal K, Lyons J, Dehzangi A, Sharma A, Wang J, et al. Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. Sci Rep 2015;5:11476.

[171] Spencer M, Eickholt J, Cheng J. A deep learning network approach to *ab initio* protein secondary structure prediction. IEEE/ACM Trans Comput Biol Bioinform 2015;12:103–12.

[172] Baldi P, Pollastri G, Andersen CAF, Brunak S. Matching protein beta-sheet partners by feedforward and recurrent neural networks. Proc Int Conf Intell Syst Mol Biol 2000:25–36.

[173] Baldi P, Brunak S, Frasconi P, Soda G, Pollastri G. Exploiting the past and the future in protein secondary structure prediction. Bioinformatics 1999;15:937–46.

[174] Pollastri G, Przybylski D, Rost B, Baldi P. Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. Proteins 2002;47:228–35.

[175] Pollastri G, Baldi P. Prediction of contact maps by GIOHMMs and recurrent neural networks using lateral propagation from all four cardinal corners. Bioinformatics 2002;18.

[176] Baldi P, Pollastri G. The principled design of large-scale recursive neural network architectures-DAG-RNNs and the protein structure prediction problem. J Mach Learn Res 2004;4:575–602.

[177] Di Lena P, Nagata K, Baldi P. Deep architectures for protein contact map prediction. Bioinformatics 2012;28:2449–57.

[178] Sønderby SK, Winther O. Protein secondary structure prediction with long short term memory networks. arXiv1412.7828

[179] Li Y, Shibuya T. Malphite: a convolutional neural network and ensemble learning based protein secondary structure predictor. Proc IEEE Int Conf Bioinformatics Biomed 2015;1260–6.

[180] Lin Z, Lanchantin J, Qi Y. MUST-CNN: a multilayer shift-and-stitch deep convolutional architecture for sequence-based protein structure prediction. Proc Conf AAAI Artif Intell 2016:8.

[181] Lena PD, Nagata K, Baldi PF. Deep spatio-temporal architectures and learning for protein structure prediction. Adv Neural Inf Process Syst 2012:512–20.

[182] Troyanskaya OG. Deep supervised and convolutional generative stochastic network for protein secondary structure prediction. Proc 31st Int Conf Mach Learn 2014; 32:745–53.

[183] Wang S, Weng S, Ma J, Tang Q. DeepCNF-D: predicting protein order/disorder regions by weighted deep convolutional neural fields. Int J Mol Sci 2015;16:17315–30.

[184] Eickholt J, Cheng J. DNdisorder: predicting protein disorder using boosting and deep networks. BMC Bioinformatics 2013;14:88.

[185] Wang S, Li W, Liu S, Xu J. RaptorX-Property: a web server for protein structure property prediction. Nucleic Acids Res 2016;44.

[186] Shin HC, Orton M, Collins DJ, Doran S, Leach MO. Autoencoder in time-series analysis for unsupervised tissues characterisation in a large unlabelled medical image dataset. Proc Int Conf Mach Learn Appl 2011;1:259–64.

[187] Jia X, Li K, Li X, Zhang A. A novel semi-supervised deep learning framework for affective state recognition on EEG signals. Proc IEEE Int Symp Bioinformatics Bioeng 2014:30–7.

[188] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2015;7:171–80.

[189] Szegedy C, Ioffe S, Vanhoucke V. Inception-v4, InceptionResNet and the impact of residual connections on learning. arXiv1602.07261.

[190] Yarlagadda DVK, Rao P, Rao D. MitosisNet: a deep learning network for mitosis detection in breast cancer histopathology images. IEEE EMBS Int Conf Biomed Health Inform 2017.

[191] Irshad H, Oh EY, Schmolze D, Quintana LM, Collins L, Tamimi RM, et al. Crowdsourcing scoring of immunohistochemistry images: evaluating performance of the crowd and an automated computational method. arXiv160606681.

[192] Albarqouni S, Baur C, Achilles F, Belagiannis V, Demirci S, Navab N. AggNet: Deep learning from crowds for mitosis detection in breast cancer histology images. IEEE Trans Med Imaging 2016;35:1313–21.

Figure legends

Figure 1 Timeline of the development of deep learning and commonly-used machine learning algorithms

The development of deep learning and neural networks is shown in the top panel, and several commonly-used machine learning algorithms are shown in the bottom panel. NN, neural network; BP, backpropagation; DBN, deep belief network; SVM, support vector machine；AE: auto-encoder; VAE: variational AE; GAN: generative adversarial network; WGAN: Wasserstein GAN.

Figure 2 Illustration of convolutional neural network

A. In the convolution layer, fields (different color blocks in the table) of the input patch (represented by a) are multiplied by matrices (convolution kernel, represented by k). **B.** In the pooling layer, the results of convolution are summarized (the max pooling is taken as example here). a_{ij}, c_{ij}, k_{ij} represent the number located in line i and column j in the corresponding matrix.

Figure 3 Illustration of recurrent neural network

A. The unfold form of common neural networks (top) and schema (bottom). **B.** An illustration of recurrent neural networks (top) and their unfold form (bottom). The red square represents one time step delay. Different from panel A, the arrows in panel B represent sets of connections. W and B represent the weight matrix and bias vector, respectively. x and y represent the input and output of the network, respectively; h indicates the hidden units of network; L consists of couples of transformations, such as densely-connected layers or dropout layers; U indicates the transformation between two neighbor time points; and t represents the time point.

Figure 4 Popularity of deep learning frameworks in Github

The distributions of stars in Github of deep learning frameworks written in C++, Lua, Python, Matlab, Julia, and Java are shown in the pie chart. More stars in Github indicate higher popularity. Font size of the frameworks in the pie chart reflects the number of stars.

Tables

Table 1 Applications of deep learning frameworks in biomedical informatics
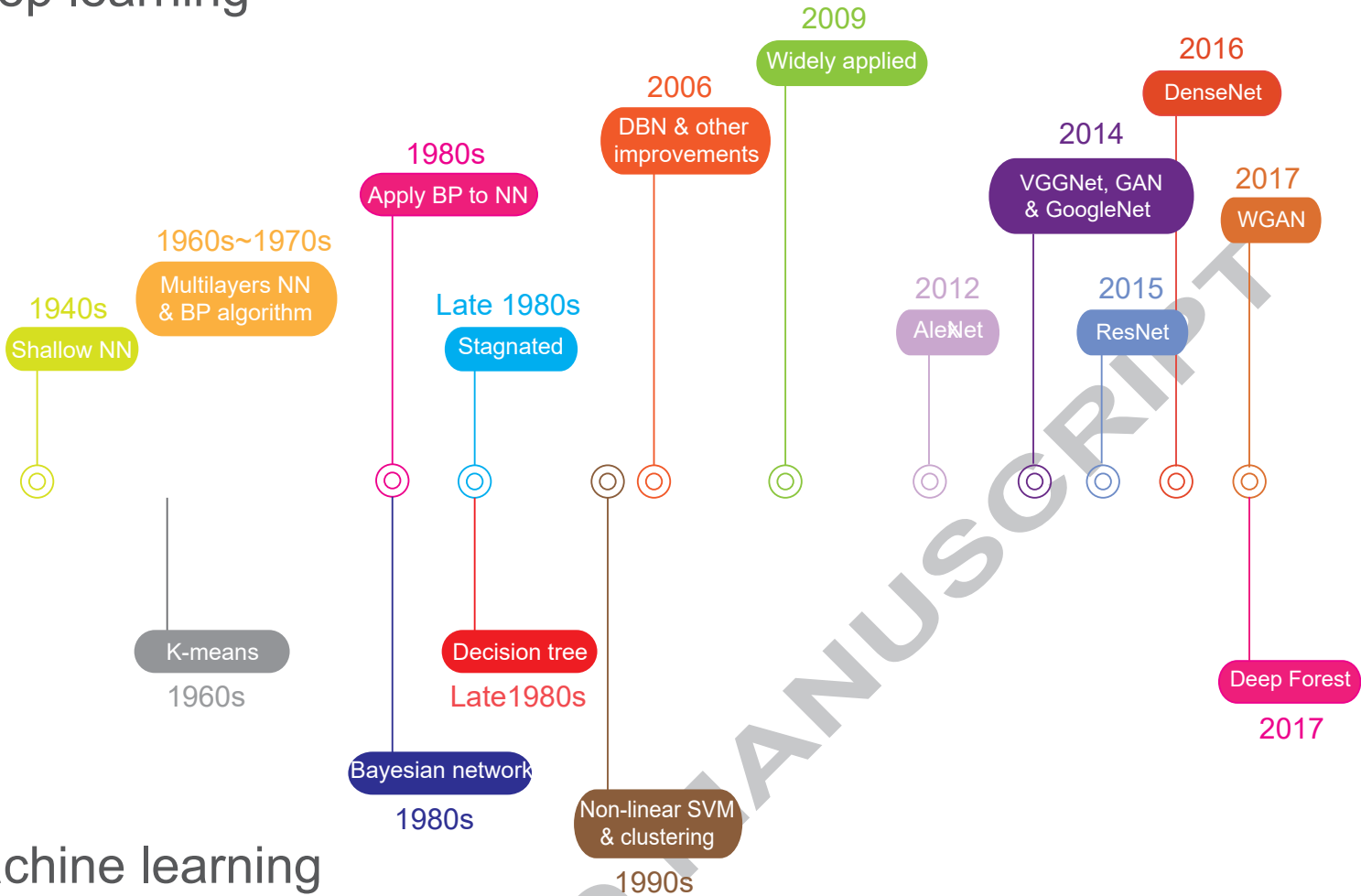
Figure 1

Deep learning
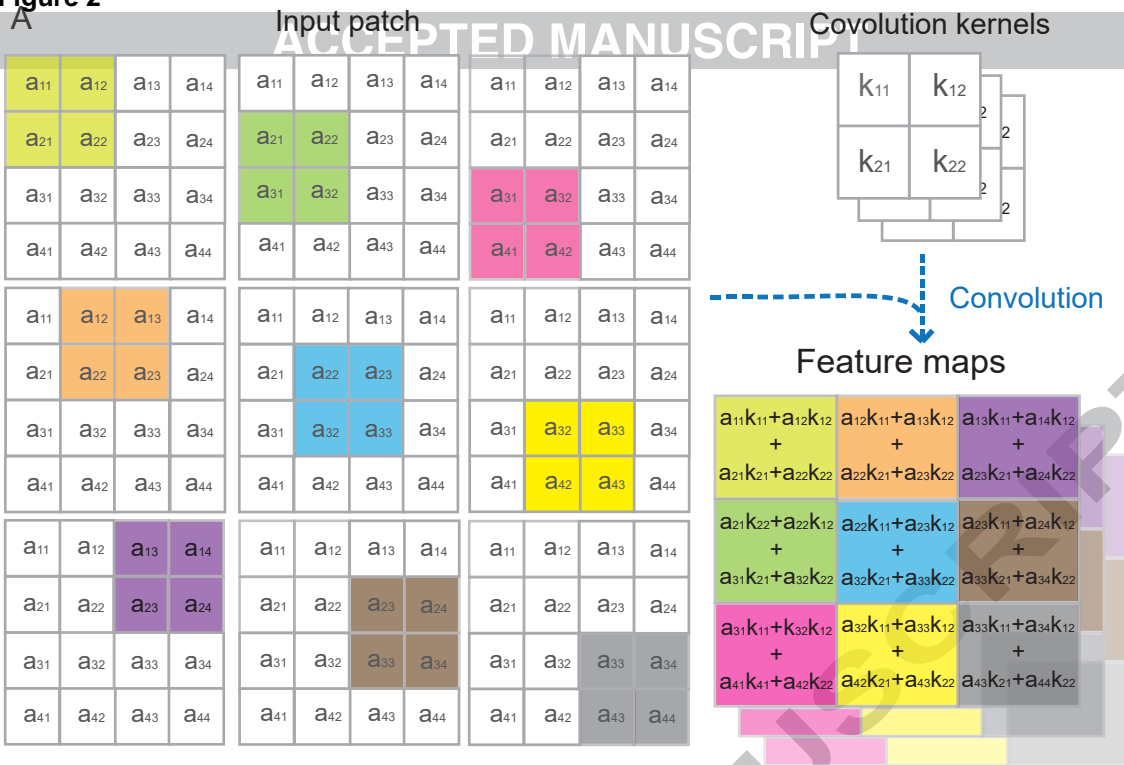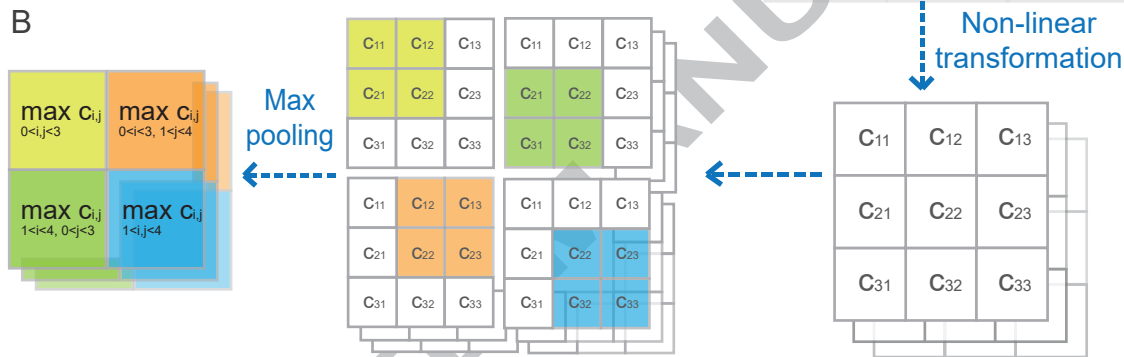
1940s
Shallow NN

1960s~1970s
Multilayers NN & BP algorithm

1980s
Apply BP to NN

Late 1980s
Stagnated

2006
DBN & other improvements

2009
Widely applied

2012
AlexNet

2014
VGGNet, GAN & GoogleNet

2015
ResNet

2016
DenseNet

2017
WGAN

K-means
1960s

Decision tree
Late1980s

Bayesian network
1980s

Non-linear SVM & clustering
1990s

Deep Forest
2017

Machine learning

Figure 2

Figure 3

A

x $\xrightarrow{\textbf{W}}$ h $\xrightarrow{\textbf{V}}$ o $\overset{L}{\cdots}$ y

h_1 $\cdots$ h_n

B

x $\xrightarrow{\textbf{W}}$ $\boxed{h}$ $\xrightarrow{\textbf{V}}$ o $\overset{L}{\cdots}$ y

$y(i)$

L

$o(i)$

$\textbf{V}$

$h(i)$ $\xrightarrow{\textbf{U}}$ $\xrightarrow{\textbf{U}}$

$\textbf{W}$

$x(i)$

$i =$ $t-1$ t $t+1$

Figure 4

Table 1

Table 1   Applications of deep learning frameworks in biomedical informatics

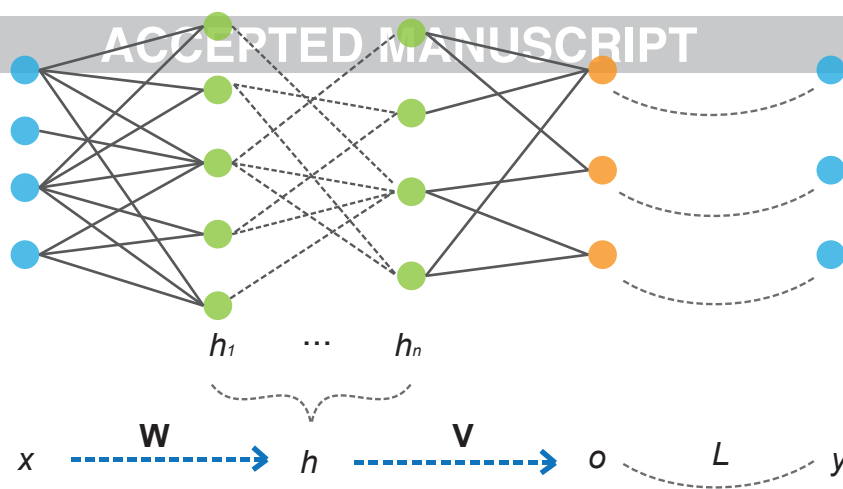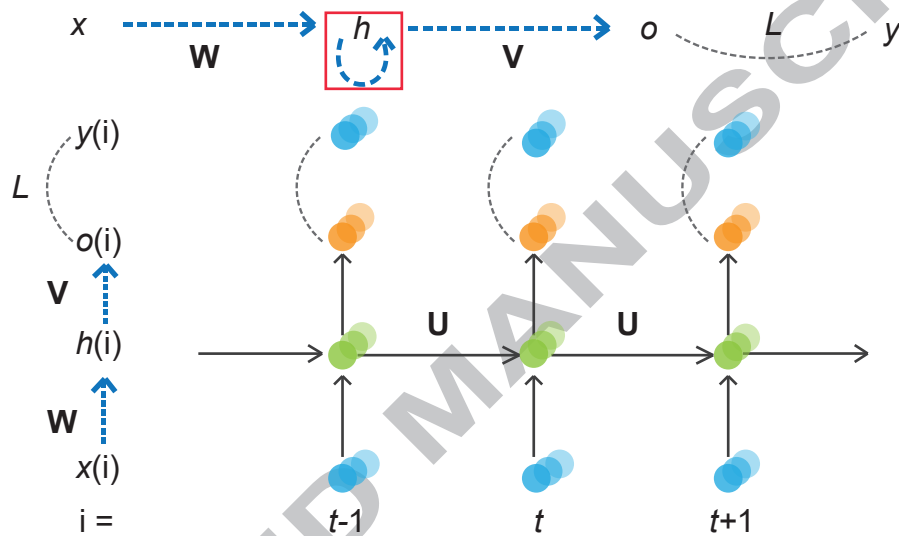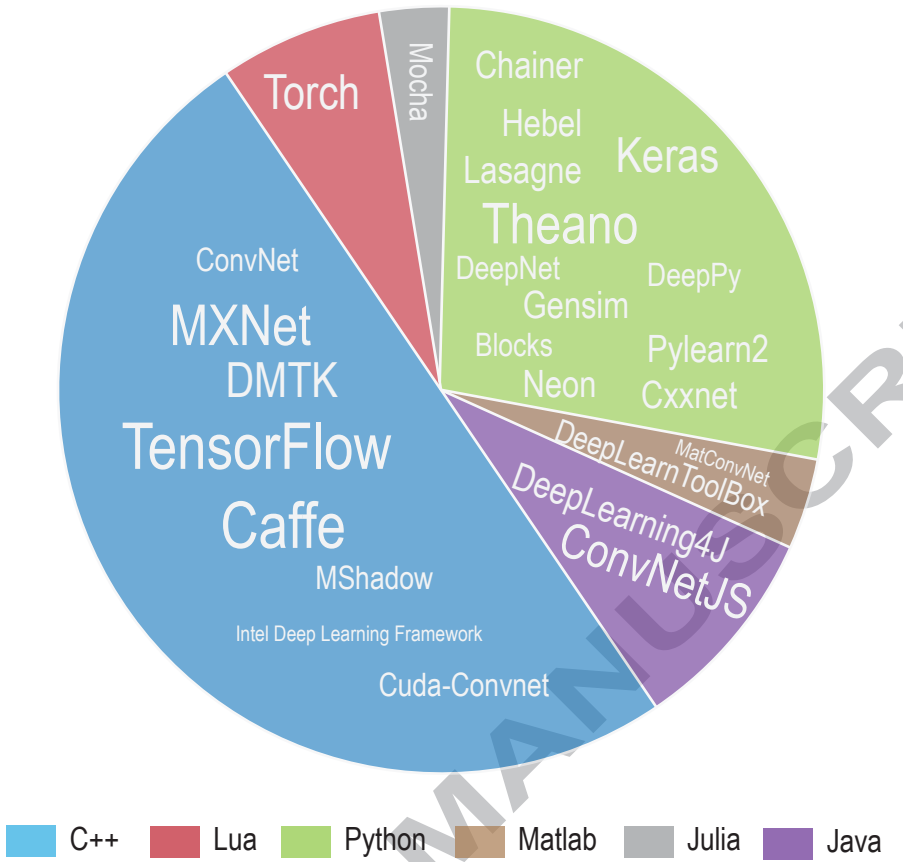| Topic	DL architecture	Brief description	Refs.
Medical images analysis	CNN	Brain tumor segmentation, won top 2 in BRATS	[108]
		Segmentation of pancreas in CT	[113]
		Knee cartilage segmentation	[114]
		Segmentation of hippocampus	[117]
		Predict semantic descriptions from medical images	[118]
		Segmentation of MR brain images	[121]
		Anatomy-specific classification of medical images	[123]
		Cerebral microbleeds from MR images	[125]
		Coronary artery calcium scoring in CT images	[126]
		Nuclei detection in routine colon cancer histology images	[129]
		Histopathological cancer classification	[130]
		Invasive ductal carcinoma segmentation in WSI	[132]
		Mammographic lesions detection	[133]
		Haemorrhages detection in fundus images	[137]
		Exudates detection in fundus images	[138]
	SAE	Segmentation of hippocampus from infant brains	[116]
		Organ detection in 4D patient data	[122]
		Histological characterization healthy skin and healing wounds	[124]
		Scoring of percentage mammographic density and mammographic texture related to breast cancer risk	[134]
		Optic disc detection from fundus photograph	[135]
	DBN	Segmentation of left ventricle of the heart from MR data	[112]
		Discriminate retinal-based diseases	[139]
	DNN	Brain tumor segmentation in MR images, won 2nd place in BRATS	[109]
		Prostate MR segmentation	[115]
		Gland instance segmentation	[119]
		Semantic segmentation of tissues in CT images	[120]
		Mitosis detection in breast cancer histological images	[131]

	RNN	EEG-based prediction of epileptic seizures propagation using time-delayed NN	[141]
		Classification of patterns of EEG synchronization for seizure prediction	[142]
		EEG-based lapse detection	[143]
		Prediction of epileptic seizures	[144]
Genomic sequencing and gene expression analysis	DNN	Gene expression inference	[145]
		Identification of *cis*-regulatory regions and replication timing domains	[151]
		Prediction of enhancer	[152]
		Prediction of splicing patterns in individual tissues and differences in splicing patterns across tissues	[159]
		Annotation of the pathogenicity of genetic variants	[161]
	DBN	Modeling structural binding preferences and predicting binding sites of RNA-binding proteins	[146]
		Prediction of splice junction at DNA level	[156]
		Prediction of transcription factor binding sites	[148,149]
		Annotation and interpretation of the noncoding genome	[151]
		Prediction of the noncoding variant effects de novo from sequence	[162]
	RNN	Prediction of miRNA precursor and miRNA targets	[153,154]
		Detection of splice junctions from DNA sequences	[157]
		Prediction of non-coding function *de novo* from sequence	[163]
		Analysis of human splicing codes and their determination of diseases	[161]
Protein structure prediction	DBN	Modeling structural binding preferences and predicting binding sites of RBPs	[146]
		Ab initio prediction of the protein secondary structures	[171]
		Prediction of protein disorder	[184]
		Prediction of secondary structures, local backbone angles, and solvent accessible surface area of proteins	[170]
	CNN	Prediction of protein order/disorder regions	[183]

	Prediction of protein secondary structures	[179−180, 182]
	Prediction of protein structure properties, including secondary structure, solvent accessibility, and disorder regions	[185]
SAE	Sequence-based prediction of backbone Cα angles and dihedrals	[169]
RNN	Prediction of protein secondary structure	[172−174, 178]
	Prediction of protein contact map	[175−177]

Note: NN, neural networks; CNN, convolutional NN; SAE, stacked auto-encoder; DBN, deep belief network; RNN, recurrent NN.