IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 6, No. 2, June 2017, pp. 66~73
ISSN: 2252-8938, DOI: 10.11591/ijai.v6.i2.pp66-73

Deep Machine Learning and Neural Networks: An Overview

Chandrahas Mishra, D. L. Gupta
Department of Computer Science & Engineering, KNIT Sultanpur2, India

Article Info

Article history:

Received Feb 10, 2017
Revised Apr 14, 2017
Accepted May 23, 2017

Keyword:

Artificial neural network (ANN).
Automatic speech recognition (ASR)
Convolutional neural networks (CNNs) and deep belief networks (DBNs)
Feature representation
Machine learning (ML)
neural nets models

ABSTRACT

Deep learning is a technique of machine learning in artificial intelligence area. Deep learning in a refined "machine learning" algorithm that far surpasses a considerable lot of its forerunners in its capacities to perceive syllables and picture. Deep learning is as of now a greatly dynamic examination territory in machine learning and example acknowledgment society. It has increased colossal triumphs in an expansive zone of utilizations, for example, speech recognition, computer vision and natural language processing and numerous industry item. Neural network is used to implement the machine learning or to design intelligent machines. In this paper brief introduction to all machine learning paradigm and application area of deep machine learning and different types of neural networks with applications is discussed.

Corresponding Author:

Chandrahas Mishra,
Department of Computer Science & Engineering,
KNIT Sultanpur2,
India.

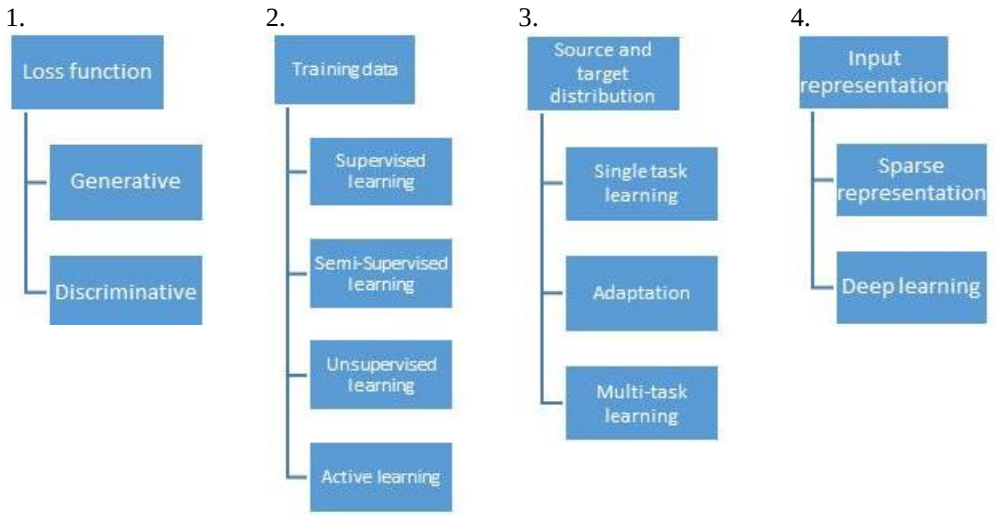
1. INTRODUCTION

1.1. Machine Learning

Learning is a process in which association of events with consequences is done. Thus basically learning is a way to substantiate the cause and effect principle. The science of designing the intelligent machine is referred to as machine learning and the tool used to design such intelligent machine is neural networks. Neural network may considered as black-box which gives some desired output for the given input. It is achieved through process called training.

In contrast to most conventional learning methods, which are considered using shallow-structured learning architectures, deep learning refers to machine learning techniques that use supervised and/or unsupervised strategies to automatically learn hierarchical representations in deep architectures for classification. Inspired by biological observations on human brain mechanisms for processing of natural signals, deep learning has attracted much attention from the academic community in recent years due to its state-of-the-art performance in many research domains such as speech recognition, collaborative filtering, and computer vision. Deep learning has additionally been effectively connected in industry items that exploit the expansive volume of advanced information. Companies like Google, Apple, and Facebook, who collect and analyse massive amounts of data on a daily basis, have been aggressively pushing forward deep learning related projects. For example, Apple's Siri, the virtual personal assistant in iPhones, offers a wide variety of services including weather reports, sport news, answers to user's questions, and reminders etc. by utilizing deep learning and more and more data collected by Apple services. Google applies deep learning algorithms to massive chunks of messy data obtained from the Internet for Google's translator.

Deep learning refers to a class of ML techniques, where many layers of information processing stages in hierarchical architectures are exploited for unsupervised feature learning and for pattern classification. It is in the intersections among the research areas of neural network, graphical modelling, optimization, pattern recognition, and signal processing. Two important reasons for the popularity of deep learning today are the significantly lowered cost of computing hardware and the drastically increased chip processing abilities (e.g., GPU units). Since 2006, researchers have demonstrated the success of deep learning in diverse applications of computer vision, phonetic recognition, voice search, spontaneous speech recognition, speech and image feature coding, semantic utterance classification, hand-writing recognition, audio processing, information retrieval, and robotics. Before going to deal with different machine learning paradigm in detail a brief classification is here. We use four key attribute to classify the machine learning paradigm.



1.2. Generative Learning

Generative learning and discriminative learning are the two most prevalent, antagonistically paired ML paradigms developed and deployed in ASR (Automatic speech recognition). There are two key factors that distinguish generative learning from discriminative learning: the nature of the model (and hence the decision function) and the loss function (i.e., the core term in the training objective). Briefly speaking, generative learning consists of

1. Using a generative model, and
2. Adopting a training objective function based on the joint likelihood loss defined on the generative model.

Discriminative learning, on the other hand, requires either

1. Using a discriminative model, or
2. Applying a discriminative training objective function to a generative model.

In this and the next sections, we will discuss generative vs. discriminative learning from both the model and loss function perspectives. While historically there has been a strong association between a model and the loss function chosen to train the model, there has been no necessary pairing of these two component in the literature

1.3. Discreminative Learning

As discussed earlier, the paradigm of discriminative learning involves either using a discriminative model or applying discriminative training to a generative model. In this section, we first provide a general discussion of the discriminative models and of the discriminative loss functions used in training, followed by an overview of the use of discriminative learning in ASR applications including its successful hybrid with generative learning.

Models:

Discriminative models make direct use of the conditional relation of labels given input vectors. One major school of such models are referred to as *Bayesian Minimum Risk* (BMR) classifiers. Shown in equation 1.

$$f(\mathbf{x};\lambda) = -\arg\min_{y'} \sum_{y} \Delta(y', y) p(y|\mathbf{x};\lambda) \tag{1}$$

Loss Functions:

This section introduces a number of discriminative loss functions. The first group of loss functions are based on probabilistic models, while the second group on the notion of *margin*.

1) Probability-Based Loss: Similar to the joint likelihood loss discussed in the preceding section on generative learning, *conditional likelihood loss* is a probability-based loss function but is defined upon the conditional relation of class labels given input features. Shown in equation 2:

$$L\left(f(\mathbf{x}), y\right) = -\ln p(y|\mathbf{x};\lambda) \tag{2}$$

This loss function is strongly tied to probabilistic discriminative models such as conditional log linear models and MLPs, while they can be applied to generative models as well, leading to a school of discriminative training methods which will be discussed shortly. Moreover, conditional likelihood loss can be naturally extended to predicting structure output. For example, when applying to Markov random fields, we obtain the training objective of conditional random fields (CRFs): by equation 3

$$p(\mathbf{y}|\mathbf{x};\lambda) = \frac{\cdot}{Z_\lambda(\mathbf{x})} \exp \lambda \cdot f(\mathbf{y}, \mathbf{x}). \tag{3}$$

Note that in most of the ML as well as the ASR literature, one often calls the training method using the conditional likelihood loss above as simply maximal likelihood estimation (MLE).

A generalization of conditional likelihood loss is Minimum Bayes Risk training. This is consistent with the criterion of MBR classifiers described in the previous subsection. The loss function of (MBR) in training is given by equation 4

$$L\left(f(\mathbf{x}), \mathbf{y}\right) = -\ln \sum_{\mathbf{y}} \Delta(\mathbf{y}', \mathbf{y}) p(\mathbf{y}|\mathbf{x};\lambda) \tag{4}$$

1.4. Semi-Supervised and Active Learning

The preceding overview of generative and discriminative ML paradigms uses the attributes of loss and decision functions to organize a multitude of ML techniques. In this section, we use a different set of attributes, namely the nature of the training data in relation to their class labels. Depending on the way that training samples are labelled or otherwise, we can classify many existing ML techniques into several separate paradigms, most of which have been in use in the ASR practice. Supervised learning assumes that *all* training samples are labelled, while unsupervised learning assumes *none*. Semi-supervised learning, as the name suggests, assumes that both labelled and unlabelled training samples are available. Supervised, unsupervised and semi-supervised learning are typically referred to under the *passive learning* setting, where labelled training samples are generated randomly according to an unknown probability distribution. In contrast, *active learning* is a setting where the learner can intelligently choose which samples to label, which we will discuss at the end of this section. In this section, we concentrate mainly on semi-supervised and active learning paradigms. This is because supervised learning is reasonably well understood and unsupervised learning does not directly aim at predicting outputs from inputs (and hence is beyond the focus of this article). We will cover these two topics only briefly.

1.4.1. Supervised Learning

In supervised learning, the training set consists of pairs of inputs and outputs drawn from a joint distribution. Using notations introduced by equation 5:

$$\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)}) | (\mathbf{x}^{(i)}, y^{(i)}) \sim p(\mathbf{x}, y)\}_{i=1}^{\cdots} \tag{5}$$

The learning objective is again empirical risk minimization with regularization, i.e. where both input data and the corresponding output labels are provided. Notice that there may exist multiple levels of label variables, notably in ASR. In this case, we should distinguish between the fully supervised case, where labels of all levels are known, the partially supervised case, where labels at certain levels are missing. In ASR, for example, it is often the case that the training set consists of waveforms and their corresponding word-level

transcriptions as the labels, while the phone-level transcriptions and time alignment information between the waveforms and the corresponding phones are missing.

1.4.2. Unsupervised Learning

In ML, unsupervised learning in general refers to learning with the input data only. This learning paradigm often aims at building representations of the input that can be used for prediction, decision making or classification, and data compression. For example, density estimation, clustering, principle component analysis and independent component analysis are all important forms of unsupervised learning. Use of vector quantization (VQ) to provide discrete inputs to ASR is one early successful application of unsupervised learning to ASR [8]. More recently, unsupervised learning has been developed as a component of staged hybrid generative-discriminative paradigm in ML. This emerging technique, based on the deep learning framework, is beginning to make impact on ASR. Learning sparse speech representations, to be talked about can likewise be viewed as unsupervised feature learning or learning feature representations in absence of classification labels.

1.4.3. Semi-Supervised Learning

The semi-supervised learning paradigm is of special significance in both theory and applications. In many ML applications including ASR, unlabelled data is abundant but labelling is expensive and time-consuming. It is possible and often helpful to leverage information from unlabelled data to influence learning. Semi-supervised learning is targeted at precisely this type of scenario, and it assumes the availability of both labelled and unlabelled data, i.e. given by equation 6:

$$\begin{aligned} &\cdot\ \mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})|(\mathbf{x}^{(i)}, y^{(i)}) \sim p(\mathbf{x}, y)\}_{i=1}^{m} \\ &\cdot\ \mathcal{U} = \{\mathbf{x}^{(i)}|\mathbf{x}^{(i)} \sim p(\mathbf{x})\}\}_{i=m+1}^{m+n} \end{aligned} \tag{6}$$

The goal is to leverage both data sources to improve learning performance. There have been a large number of semi-supervised learning algorithms proposed in the literature and various ways of grouping these approaches. Here we categorize semi-supervised learning methods based on their inductive or transductive nature. The key difference between inductive and transductive learning is the outcome of learning process. In the former setting, the goal is to find a decision function that not only correctly classifies training set samples, but also generalizes to any future sample. In contrast, transductive learning aims at directly predicting the output labels of a test set, without the need of generalizing to other samples. In this regard, the direct outcome of transductive semi-supervised learning is a set of labels instead of a decision function. All learning paradigms we have presented are inductive in nature.

1.4.4. Active Learning

Active learning is a similar setting to semi-supervised learning in that, in addition to a small amount of labelled data, there is a large amount of unlabelled data available; i.e., given by equation 7:

$$\begin{aligned} &\cdot\ \mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})|(\mathbf{x}^{(i)}, y^{(i)}) \sim p(\mathbf{x}, y)\}_{i=1}^{m} \\ &\cdot\ \mathcal{U} = \{\mathbf{x}^{(i)}|\mathbf{x}^{(i)} \sim p(\mathbf{x})\}\}_{i=m+1}^{m+n} \end{aligned} \tag{7}$$

The goal of active learning, however, is to query the most informative set of inputs to be labelled, hoping to improve classification performance with the minimum number of queries. That is, in active learning, the learner may play an active role in deciding the data set rather than it be passively given. The key thought behind active learning is that a ML calculation can accomplish more noteworthy execution, e.g., higher classification accuracy, with fewer training labels if it is allowed to choose the subset of data that has labels. An active learner might posture questions ordinarily as unlabelled information cases to be named regularly by a human. For this reason, it is sometimes called query learning. Active learning is all around inspired in numerous present day ML issues where unlabelled information might be copious or effortlessly gotten yet names are troublesome tedious, or expensive to obtain. This is the situation for speech recognition. Broadly, active learning comes in two forms: batch active learning, where a subset of data is chosen, a priori in a batch to be labelled. The labels of the instances in the batch chosen to be labelled may not, under this approach, influence other instances to be selected since all instances are chosen at once. In online active learning, on the other hand, instances are chosen one-by-one, and the true labels of all previously labelled instances may be used to select other instances to be labelled. For this reason, online active learning is sometimes considered more powerful.

1.5. Artificial Neural Network

An artificial neural network is an interconnected group of nodes, distantly related to the vast network of neurons in a brain shown in Figure 1. Here, each circular node represents an artificial neuron and an arrow represents a connection from the output of one neuron to the input of another should (ideally) be able to handle this. Artificial neural network consist of three type of layer namely input layer, hidden layer and output layer. Hidden layer is connected between input and output layer.

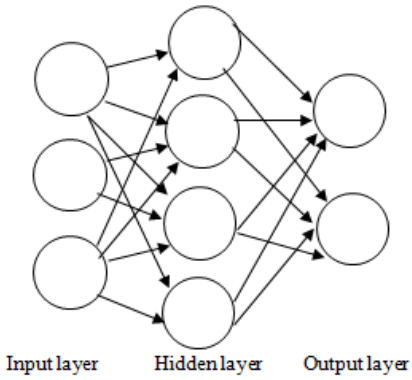


Figure 1. Artificial Neural Network Architecture

1.6. Convolutional Neural Networks

CNNs are a family of multi-layer neural networks shown in Figure 2 particularly designed for use on two-dimensional data, such as images and videos. CNNs are influenced by earlier work in time-delay neural networks (TDNN), which reduce learning computation requirements by sharing weights in a temporal dimension and are intended for speech and time-series processing. CNNs are the first truly successful deep learning approach where many layers of a hierarchy were successfully trained in a robust manner. A CNN is a choice of architecture that leverages spatial and temporal relationships to reduce the number of parameters which must be learned and thus improves upon general feed-forward back propagation training. CNNs were proposed as a deep learning framework that is motivated by minimal data pre-processing requirements. In CNNs, small portions of the image are treated as inputs to the lowest layer of the hierarchical structure. Information generally propagates through the different layers of the network whereby at each layer digital filtering is applied in order to obtain salient features of the data observed. The method provides a level of invariance to shift, scale and rotation as the local receptive field allows the neuron or processing unit access to elementary features such as oriented edges or corners.

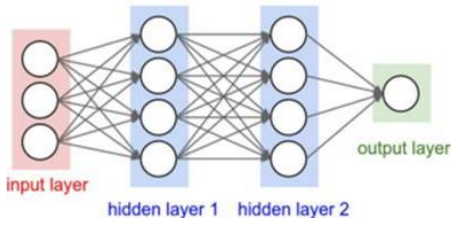


Figure 2. Convolutional Neural Network (CNN) Architecture [12]

1.7. Deep Belief Networks

DBNs are composed of several layers of Restricted Boltzmann Machines, a type of neural network shown in Figure 3. These networks are “restricted” to a single visible layer and single hidden layer, where connections are formed between the layers (units within a layer are not connected). The hidden units are trained to capture higher-order data correlations that are observed at the visible units. Initially, aside from the top two layers, which form an associative memory, directed top-down generative weight are used to connect the layers of a DBN. RBMs are attractive as a building block, over more traditional and deeply layered

sigmoid belief networks, due to their ease of learning these connection weights, the initial pre-training occurs in an unsupervised greedy layer-by-layer manner to obtain generative weights, enabled by what Hinton has termed contrastive divergence. During this training phase, a vector v is presented to the visible units that forward values to the hidden units. Going in reverse, the visible unit inputs are then stochastically found in an attempt to reconstruct the original input. Finally, these new visible neuron activations are forwarded such that one step reconstruction hidden unit activations, h, can be attained. Performing these back and forth steps is a process known as Gibbs sampling, and the difference in the correlation of the hidden activations and visible inputs forms the basis for a weight update. Training time is significantly reduced as it can be shown that only a single step is needed to approximate maximum likelihood learning. Each layer added to the network can improves the log probability of the training data, which we can be think of as increasing true representational power of network. This meaningful expansion, in conjunction with the utilization of unlabelled data, is a critical component in any deep learning application.

At the top two layers, the weights are tied together, such that the output of the lower layers provides a reference clue or link for the top layer to "associate" with its memory contents. We often encounter problems where discriminative performance is of ultimate concern, e.g. in classification tasks. A DBN may be fine-tuned after pre-training for improved discriminative performance by utilizing labelled data through back-propagation. At this point, a set of labels is attached to the top layer (expanding the associative memory) to clarify category boundaries in the network through which a new set of bottom-up, recognition weights are learned. It has been shown in that such networks often perform better than those trained exclusively with backpropagation. This may be intuitively explained by the fact that back-propagation

For DBNs is only required to perform a local search on the weight (parameter) space, speeding training and convergence time in relation to traditional feed-forward neural networks

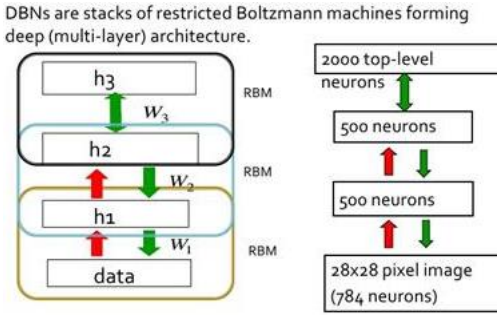


Figure 3. Deep Belief Neural Network (DBNN) Architecture [13]

2. IMPLEMENTATION TECHNIQUES

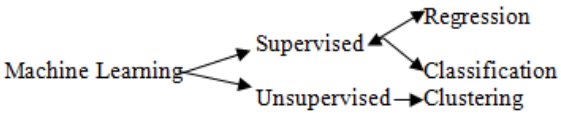
Learning can be implemented by various methodology or techniques depending upon the requirement. Basically learning can be categorized in two types:

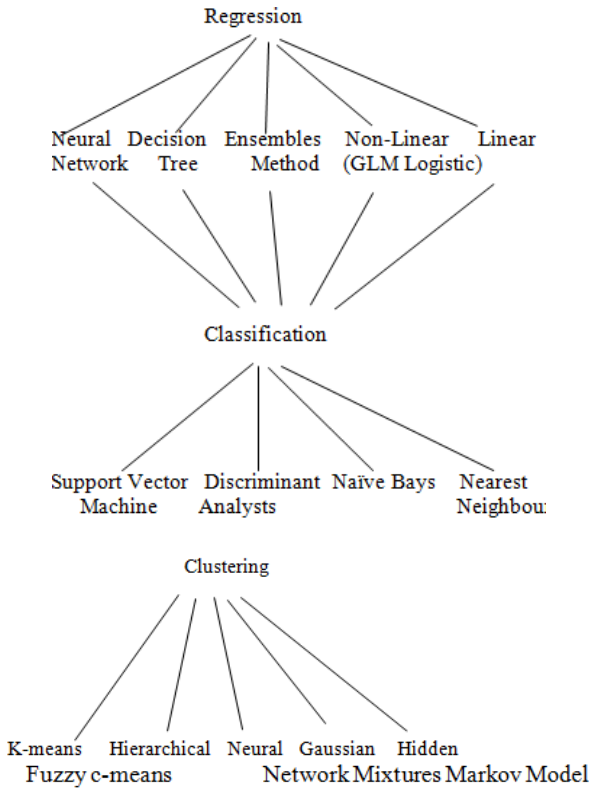
1. Supervised learning
2. Unsupervised learning

In supervised learning two technique is used

1. Regression
2. Classification

While unsupervised learning is implemented by clustering algorithms





3. APPLICATIONS

According to E. Don Box et al. [1] neural networks model can be used for demand forecasting in deregulated environment. Neural networks are designed and trained on the basis of aggregate demands of the groups of surveyed customers of different categories.

The most frequently encountered decision making tasks of human activity is classification. Classification problem occurs when an object needs to be assigned into a predefined class based on a number of observed attributes related to that object. Many problems in business, science, industry, and medicine can be treated as classification problems. Examples include bankruptcy prediction, credit scoring, medical diagnosis, quality control, handwritten character recognition, and speech recognition. [2]

Neural network and genetic algorithms are used for web mining. Sankar K. Pal et.al [3] describe web mining in soft computing framework.

Soft computing paradigm like fuzzy sets (FS), artificial neural networks (ANN) and support vector machines (SVMs) is used in Bioinformatics. [4]

The research community has statrted looking for IP traffic classification techniques that do not rely on ‘well known’ TCP or UDP port numbers, or interpreting the contents of packet payloads. New work is emerging on the use of statistical traffic characteristics to assist in the identification and classification process. This survey paper looks at emerging research into the application of Machine Learning (ML) techniques to IP traffic classification - an inter-disciplinary blend of IP networking and data mining techniques [5]

It is crucial for financial institutions, the ability to predict or forecast business failures, as incorrect decisions can have direct financial consequences. There are the two major research problems in the accounting and finance domain are Bankruptcy prediction and credit scoring. In the literature, a number of models have been developed to predict whether borrowers are in danger of bankruptcy and whether they should be considered a good or bad credit risk. Since the 1990s, machine-learning techniques, such as neural networks and decision trees, have been studied extensively as tools for bankruptcy prediction and credit score modelling. [6]

Learning methods that have been applied to CRs classifying them under supervised and unsupervised learning. Some of the most important, and commonly used, learning algorithms was provided along with their advantages and disadvantages are discussed in this literature. [7]

4. CONCLUSION

In this paper a deep discussion about machine learning methods and their implementation has been discussed. It is clearly shown that different methods uses different algorithm for implementation. It is also concluded that Neural Network and Support vector machine is most popular techniques to implement the machine learning paradigm. Deep learning is extended version of supervised learning. It is finally concluded that Convolution neural network and Deep Belief network are two powerful techniques which may be used to solve various complex problems using deep learning. Deep learning platforms can also be benefited from engineered features while learning more complex representations which engineered systems typically lack. It is abundantly clear that advancements made with respect to developing deep machine learning systems will undoubtedly shape the future of machine learning and artificial intelligence systems in general.

REFERENCES

[1] Charytoniuk Wiktor, Box E. Don, Lee Wei-Jen, Mo-Shing, Kotas Chen Paul and Olinda Peter Van, "Neural-Network-Based Demand Forecasting in Deregulated Environment," *IEEE Trans. On Industry Applications, Vol. 36, No. 3, May/June 20*00.

[2] Zhang Guoqiang Peter, "Neural Networks for Classification: A Survey," *IEEE Trans. On Systems, Man, And Cybernatics—Part C: Applications and reviews, Vol. 30, No. 4, November* 2000.

[3] Sankar K. Pal, Varun Talwar, and Pabitra Mitra, "Web Mining in Soft Computing Framework: Relevance, State of the Art and Future Directions,*" IEEE Trans.On Neural Networks, Vol. 13, NO. 5, September 2002*.

[4] Sushmita Mitra, Yoichi Hayashi, "Bioinformatics with Soft Computing,"*IEEE Trans. On System, Man and Cybernatics—Part C: Application and Reviews, Vol. 36, No. 5, September* 2006.

[5] Thuy T.T. Nguyen and Grenville Armitage,"A Survey of Techniques for Internet Traffic Classification using Machine Learning," *IEEE Communications Surveys & Tutorials, Vol. 10, No. 4, Fourth Quarter* 2008.

[6] Wei-Yang Lin, Ya-Han Hu Chih-Fong Tsai, "Machine Learning in Financial Crisis Prediction: A Survey," *IEEE Trans. On System, Man and Cybernatics—Part C: Application and Reviews, Vol. 42, No. 4, July* 2012.

[7] Mario Bkassiny Yang Li, Sudharman K. Jayaweera, "A Survey on Machine-Learning Techniques in Cognitive Radios," *IEEE Communications Surveys & Tutorials*, Vol. 15, No. 3, Third Quarter 2013.

[8] Li Deng, Xiao Li, "Machine Learning Paradigms for Speech Recognition: An Overview," *IEEE Trans On Audio, Speech, And Language Processing, Vol. 21, NO. 5, May* 2013.

[9] Daizhan Cheng, Hongsheng Qi, "State–Space Analysis of Boolean Networks," *IEEE Trans.On Neural Networks, Vol. 21, No. 4, April* 2010.

[10] Yoshua Bengio, Aaron Courville and Pascal Vincent, "Representation Learning: A Review and New Perspectives,*" IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol*. 35, *NO. 8, August* 2013.

[11] Xue-Wen Chen, Xiaotong Lin, "Big Data Deep Learning: Challenges and Perspectives," *Digital Object Identifier 10.1109/IEEE ACCESS.2014.2325029.*

[12] http://cs231n.github.io/convolutional-networks/#overview

[13] http://image.slidesharecdn.com/deep-belief-nets1166/95/deep-belief-nets-3-728.jpg?cb=1272282825

APPENDIX

Definitions of a Subset of Commonly Used Symbols and Notations in This Article

Symbol	Meaning
$\mathcal{X}$	Space of input vectors
$\mathcal{Y}$	Set of output labels
$p(\mathbf{x}, y)$	Joint distribution $p(\mathbf{X} = \mathbf{x}, Y = y)$
$\mathcal{F}$	Space of decision functions $f : \mathcal{X} \to \mathcal{Y}$
$f(\mathbf{x}; \lambda)$	Decision function
$d_y(\mathbf{x}; \lambda)$	Discriminant function
λ	Model or decision function parameters
$L(f(\mathbf{x}), y)$	Loss function
$\mathrm{E}_{p(\mathbf{x},y)}[\cdot]$	Expectation $\mathrm{E}_{(\mathbf{x},y)\sim p(\mathbf{x},y)}[\cdot]$
$\mathcal{D}$	Training data $(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})_{i=1}^{m}$