3

Three Classes of Deep Learning Networks

3.1 A three-way categorization

As described earlier, deep learning refers to a rather wide class of machine learning techniques and architectures, with the hallmark of using many layers of non-linear information processing that are hierarchical in nature. Depending on how the architectures and techniques are intended for use, e.g., synthesis/generation or recognition/classification, one can broadly categorize most of the work in this area into three major classes:

1. **Deep networks for unsupervised or generative learning**, which are intended to capture high-order correlation of the observed or visible data for pattern analysis or synthesis purposes when no information about target class labels is available. Unsupervised feature or representation learning in the literature refers to this category of the deep networks. When used in the generative mode, may also be intended to characterize joint statistical distributions of the visible data and their associated classes when available and being treated as part of the visible data. In the

latter case, the use of Bayes rule can turn this type of generative networks into a discriminative one for learning.

2. **Deep networks for supervised learning**, which are intended to directly provide discriminative power for pattern classification purposes, often by characterizing the posterior distributions of classes conditioned on the visible data. Target label data are always available in direct or indirect forms for such supervised learning. They are also called discriminative deep networks.
3. **Hybrid deep networks**, where the goal is discrimination which is assisted, often in a significant way, with the outcomes of generative or unsupervised deep networks. This can be accomplished by better optimization or/and regularization of the deep networks in category (2). The goal can also be accomplished when discriminative criteria for supervised learning are used to estimate the parameters in any of the deep generative or unsupervised deep networks in category (1) above.

Note the use of "hybrid" in (3) above is different from that used sometimes in the literature, which refers to the hybrid systems for speech recognition feeding the output probabilities of a neural network into an HMM [17, 25, 42, 261].

By the commonly adopted machine learning tradition (e.g., Chapter 28 in [264], and Reference [95], it may be natural to just classify deep learning techniques into deep discriminative models (e.g., deep neural networks or DNNs, recurrent neural networks or RNNs, convolutional neural networks or CNNs, etc.) and generative/unsupervised models (e.g., restricted Boltzmann machine or RBMs, deep belief networks or DBNs, deep Boltzmann machines (DBMs), regularized autoencoders, etc.). This two-way classification scheme, however, misses a key insight gained in deep learning research about how generative or unsupervised-learning models can greatly improve the training of DNNs and other deep discriminative or supervised-learning models via better regularization or optimization. Also, deep networks for unsupervised learning may not necessarily need to be probabilistic or be able to meaningfully sample from the model (e.g., traditional autoencoders, sparse coding networks, etc.). We note here that more recent

studies have generalized the traditional denoising autoencoders so that they can be efficiently sampled from and thus have become generative models [5, 24, 30]. Nevertheless, the traditional two-way classification indeed points to several key differences between deep networks for unsupervised and supervised learning. Compared between the two, deep supervised-learning models such as DNNs are usually more efficient to train and test, more flexible to construct, and more suitable for end-to-end learning of complex systems (e.g., no approximate inference and learning such as loopy belief propagation). On the other hand, the deep unsupervised-learning models, especially the probabilistic generative ones, are easier to interpret, easier to embed domain knowledge, easier to compose, and easier to handle uncertainty, but they are typically intractable in inference and learning for complex systems. These distinctions are retained also in the proposed three-way classification which is hence adopted throughout this monograph.

Below we review representative work in each of the above three categories, where several basic definitions are summarized in Table 3.1. Applications of these deep architectures, with varied ways of learning including supervised, unsupervised, or hybrid, are deferred to Sections 7–11.

3.2 Deep networks for unsupervised or generative learning

Unsupervised learning refers to no use of task specific supervision information (e.g., target class labels) in the learning process. Many deep networks in this category can be used to meaningfully generate samples by sampling from the networks, with examples of RBMs, DBNs, DBMs, and generalized denoising autoencoders [23], and are thus generative models. Some networks in this category, however, cannot be easily sampled, with examples of sparse coding networks and the original forms of deep autoencoders, and are thus not generative in nature.

Among the various subclasses of generative or unsupervised deep networks, the energy-based deep models are the most common [28, 20, 213, 268]. The original form of the deep autoencoder [28, 100, 164], which we will give more detail about in Section 4, is a typical example

Table 3.1: Basic deep learning terminologies.

Deep Learning: a class of machine learning techniques, where many layers of information processing stages in hierarchical supervised architectures are exploited for unsupervised feature learning and for pattern analysis/classification. The essence of deep learning is to compute hierarchical features or representations of the observational data, where the higher-level features or factors are defined from lower-level ones. The family of deep learning methods have been growing increasingly richer, encompassing those of neural networks, hierarchical probabilistic models, and a variety of unsupervised and supervised feature learning algorithms.
Deep belief network (DBN): probabilistic generative models composed of multiple layers of stochastic, hidden variables. The top two layers have undirected, symmetric connections between them. The lower layers receive top-down, directed connections from the layer above.
Boltzmann machine (BM): a network of symmetrically connected, neuron-like units that make stochastic decisions about whether to be on or off.
Restricted Boltzmann machine (RBM): a special type of BM consisting of a layer of visible units and a layer of hidden units with no visible-visible or hidden-hidden connections.
Deep neural network (DNN): a multilayer perceptron with many hidden layers, whose weights are fully connected and are often (although not always) initialized using either an unsupervised or a supervised pretraining technique. (In the literature prior to 2012, a DBN was often used incorrectly to mean a DNN.)
Deep autoencoder: a "discriminative" DNN whose output targets are the data input itself rather than class labels; hence an unsupervised learning model. When trained with a denoising criterion, a deep autoencoder is also a generative model and can be sampled from.

(*Continued*)

Table 3.1: (*Continued*)

Distributed representation: an internal representation of the observed data in such a way that they are modeled as being explained by the interactions of many hidden factors. A particular factor learned from configurations of other factors can often generalize well to new configurations. Distributed representations naturally occur in a "connectionist" neural network, where a concept is represented by a pattern of activity across a number of units and where at the same time a unit typically contributes to many concepts. One key advantage of such many-to-many correspondence is that they provide robustness in representing the internal structure of the data in terms of graceful degradation and damage resistance. Another key advantage is that they facilitate generalizations of concepts and relations, thus enabling reasoning abilities.

of this unsupervised model category. Most other forms of deep autoencoders are also unsupervised in nature, but with quite different properties and implementations. Examples are transforming autoencoders [160], predictive sparse coders and their stacked version, and de-noising autoencoders and their stacked versions [376].

Specifically, in de-noising autoencoders, the input vectors are first corrupted by, for example, randomly selecting a percentage of the inputs and setting them to zeros or adding Gaussian noise to them. Then the parameters are adjusted for the hidden encoding nodes to reconstruct the original, uncorrupted input data using criteria such as mean square reconstruction error and KL divergence between the original inputs and the reconstructed inputs. The encoded representations transformed from the uncorrupted data are used as the inputs to the next level of the stacked de-noising autoencoder.

Another prominent type of deep unsupervised models with generative capability is the deep Boltzmann machine or DBM [131, 315, 316, 348]. A DBM contains many layers of hidden variables, and has no connections between the variables within the same layer. This is a special case of the general Boltzmann machine (BM), which is a network of

symmetrically connected units that are on or off based on a stochastic mechanism. While having a simple learning algorithm, the general BMs are very complex to study and very slow to train. In a DBM, each layer captures complicated, higher-order correlations between the activities of hidden features in the layer below. DBMs have the potential of learning internal representations that become increasingly complex, highly desirable for solving object and speech recognition problems. Further, the high-level representations can be built from a large supply of unlabeled sensory inputs and very limited labeled data can then be used to only slightly fine-tune the model for a specific task at hand.

When the number of hidden layers of DBM is reduced to one, we have restricted Boltzmann machine (RBM). Like DBM, there are no hidden-to-hidden and no visible-to-visible connections in the RBM. The main virtue of RBM is that via composing many RBMs, many hidden layers can be learned efficiently using the feature activations of one RBM as the training data for the next. Such composition leads to deep belief network (DBN), which we will describe in more detail, together with RBMs, in Section 5.

The standard DBN has been extended to the factored higher-order Boltzmann machine in its bottom layer, with strong results obtained for phone recognition [64] and for computer vision [296]. This model, called the mean-covariance RBM or mcRBM, recognizes the limitation of the standard RBM in its ability to represent the covariance structure of the data. However, it is difficult to train mcRBMs and to use them at the higher levels of the deep architecture. Further, the strong results published are not easy to reproduce. In the architecture described by Dahl et al. [64], the mcRBM parameters in the full DBN are not fine-tuned using the discriminative information, which is used for fine tuning the higher layers of RBMs, due to the high computational cost. Subsequent work showed that when speaker adapted features are used, which remove more variability in the features, mcRBM was not helpful [259].

Another representative deep generative network that can be used for unsupervised (as well as supervised) learning is the sum–product network or SPN [125, 289]. An SPN is a directed acyclic graph with the observed variables as leaves, and with sum and product operations as internal nodes in the deep network. The "sum" nodes give mixture

models, and the "product" nodes build up the feature hierarchy. Properties of "completeness" and "consistency" constrain the SPN in a desirable way. The learning of SPNs is carried out using the EM algorithm together with back-propagation. The learning procedure starts with a dense SPN. It then finds an SPN structure by learning its weights, where zero weights indicate removed connections. The main difficulty in learning SPNs is that the learning signal (i.e., the gradient) quickly dilutes when it propagates to deep layers. Empirical solutions have been found to mitigate this difficulty as reported in [289]. It was pointed out in that early paper that despite the many desirable generative properties in the SPN, it is difficult to fine tune the parameters using the discriminative information, limiting its effectiveness in classification tasks. However, this difficulty has been overcome in the subsequent work reported in [125], where an efficient BP-style discriminative training algorithm for SPN was presented. Importantly, the standard gradient descent, based on the derivative of the conditional likelihood, suffers from the same gradient diffusion problem well known in the regular DNNs. The trick to alleviate this problem in learning SPNs is to replace the marginal inference with the most probable state of the hidden variables and to propagate gradients through this "hard" alignment only. Excellent results on small-scale image recognition tasks were reported by Gens and Domingo [125].

Recurrent neural networks (RNNs) can be considered as another class of deep networks for unsupervised (as well as supervised) learning, where the depth can be as large as the length of the input data sequence. In the unsupervised learning mode, the RNN is used to predict the data sequence in the future using the previous data samples, and no additional class information is used for learning. The RNN is very powerful for modeling sequence data (e.g., speech or text), but until recently they had not been widely used partly because they are difficult to train to capture long-term dependencies, giving rise to gradient vanishing or gradient explosion problems which were known in early 1990s [29, 167]. These problems can now be dealt with more easily [24, 48, 85, 280]. Recent advances in Hessian-free optimization [238] have also partially overcome this difficulty using approximated second-order information or stochastic curvature estimates. In the more recent work [239], RNNs

that are trained with Hessian-free optimization are used as a generative deep network in the character-level language modeling tasks, where gated connections are introduced to allow the current input characters to predict the transition from one latent state vector to the next. Such generative RNN models are demonstrated to be well capable of generating sequential text characters. More recently, Bengio et al. [22] and Sutskever [356] have explored variations of stochastic gradient descent optimization algorithms in training generative RNNs and shown that these algorithms can outperform Hessian-free optimization methods. Mikolov et al. [248] have reported excellent results on using RNNs for language modeling. Most recently, Mesnil et al. [242] and Yao et al. [403] reported the success of RNNs in spoken language understanding. We will review this set of work in Section 8.

There has been a long history in speech recognition research where human speech production mechanisms are exploited to construct dynamic and deep structure in probabilistic generative models; for a comprehensive review, see the monograph by Deng [76]. Specifically, the early work described in [71, 72, 83, 84, 99, 274] generalized and extended the conventional shallow and conditionally independent HMM structure by imposing dynamic constraints, in the form of polynomial trajectory, on the HMM parameters. A variant of this approach has been more recently developed using different learning techniques for time-varying HMM parameters and with the applications extended to speech recognition robustness [431, 416]. Similar trajectory HMMs also form the basis for parametric speech synthesis [228, 326, 439, 438]. Subsequent work added a new hidden layer into the dynamic model to explicitly account for the target-directed, articulatory-like properties in human speech generation [45, 73, 74, 83, 96, 75, 90, 231, 232, 233, 251, 282]. More efficient implementation of this deep architecture with hidden dynamics is achieved with non-recursive or finite impulse response (FIR) filters in more recent studies [76, 107, 105]. The above deep-structured generative models of speech can be shown as special cases of the more general dynamic network model and even more general dynamic graphical models [35, 34]. The graphical models can comprise many hidden layers to characterize the complex relationship between the variables in speech generation. Armed with powerful graphical

modeling tool, the deep architecture of speech has more recently been successfully applied to solve the very difficult problem of single-channel, multi-talker speech recognition, where the mixed speech is the visible variable while the un-mixed speech becomes represented in a new hidden layer in the deep generative architecture [301, 391]. Deep generative graphical models are indeed a powerful tool in many applications due to their capability of embedding domain knowledge. However, they are often used with inappropriate approximations in inference, learning, prediction, and topology design, all arising from inherent intractability in these tasks for most real-world applications. This problem has been addressed in the recent work of Stoyanov et al. [352], which provides an interesting direction for making deep generative graphical models potentially more useful in practice in the future. An even more drastic way to deal with this intractability was proposed recently by Bengio et al. [30], where the need to marginalize latent variables is avoided altogether.

The standard statistical methods used for large-scale speech recognition and understanding combine (shallow) hidden Markov models for speech acoustics with higher layers of structure representing different levels of natural language hierarchy. This combined hierarchical model can be suitably regarded as a deep generative architecture, whose motivation and some technical detail may be found in Section 7 of the recent monograph [200] on "Hierarchical HMM" or HHMM. Related models with greater technical depth and mathematical treatment can be found in [116] for HHMM and [271] for Layered HMM. These early deep models were formulated as directed graphical models, missing the key aspect of "distributed representation" embodied in the more recent deep generative networks of the DBN and DBM discussed earlier in this chapter. Filling in this missing aspect would help improve these generative models.

Finally, dynamic or temporally recursive generative models based on neural network architectures can be found in [361] for human motion modeling, and in [344, 339] for natural language and natural scene parsing. The latter model is particularly interesting because the learning algorithms are capable of automatically determining the optimal model structure. This contrasts with other deep architectures such as DBN

where only the parameters are learned while the architectures need to be pre-defined. Specifically, as reported in [344], the recursive structure commonly found in natural scene images and in natural language sentences can be discovered using a max-margin structure prediction architecture. It is shown that the units contained in the images or sentences are identified, and the way in which these units interact with each other to form the whole is also identified.

3.3 Deep networks for supervised learning

Many of the discriminative techniques for supervised learning in signal and information processing are shallow architectures such as HMMs [52, 127, 147, 186, 188, 290, 394, 418] and conditional random fields (CRFs) [151, 155, 281, 400, 429, 446]. A CRF is intrinsically a shallow discriminative architecture, characterized by the linear relationship between the input features and the transition features. The shallow nature of the CRF is made most clear by the equivalence established between the CRF and the discriminatively trained Gaussian models and HMMs [148]. More recently, deep-structured CRFs have been developed by stacking the output in each lower layer of the CRF, together with the original input data, onto its higher layer [428]. Various versions of deep-structured CRFs are successfully applied to phone recognition [410], spoken language identification [428], and natural language processing [428]. However, at least for the phone recognition task, the performance of deep-structured CRFs, which are purely discriminative (non-generative), has not been able to match that of the hybrid approach involving DBN, which we will take on shortly.

Morgan [261] gives an excellent review on other major existing discriminative models in speech recognition based mainly on the traditional neural network or MLP architecture using back-propagation learning with random initialization. It argues for the importance of both the increased width of each layer of the neural networks and the increased depth. In particular, a class of deep neural network models forms the basis of the popular "tandem" approach [262], where the output of the discriminatively learned neural network is treated as part

of the observation variable in HMMs. For some representative recent work in this area, see [193, 283].

In more recent work of [106, 110, 218, 366, 377], a new deep learning architecture, sometimes called deep stacking network (DSN), together with its tensor variant [180, 181] and its kernel version [102], are developed that all focus on discrimination with scalable, parallelizable, block-wise learning relying on little or no generative component. We will describe this type of discriminative deep architecture in detail in Section 6.

As discussed in the preceding section, recurrent neural networks (RNNs) have been used as a generative model; see also the neural predictive model [87] with a similar "generative" mechanism. RNNs can also be used as a discriminative model where the output is a label sequence associated with the input data sequence. Note that such discriminative RNNs or sequence models were applied to speech a long time ago with limited success. In [17], an HMM was trained jointly with the neural networks, with a discriminative probabilistic training criterion. In [304], a separate HMM was used to segment the sequence during training, and the HMM was also used to transform the RNN classification results into label sequences. However, the use of the HMM for these purposes does not take advantage of the full potential of RNNs.

A set of new models and methods were proposed more recently in [133, 134, 135, 136] that enable the RNNs themselves to perform sequence classification while embedding the long-short-term memory into the model, removing the need for pre-segmenting the training data and for post-processing the outputs. Underlying this method is the idea of interpreting RNN outputs as the conditional distributions over all possible label sequences given the input sequences. Then, a differentiable objective function can be derived to optimize these conditional distributions over the correct label sequences, where the segmentation of the data is performed automatically by the algorithm. The effectiveness of this method has been demonstrated in handwriting recognition tasks and in a small speech task [135, 136] to be discussed in more detail in Section 7 of this monograph.

Another type of discriminative deep architecture is the convolutional neural network (CNN), in which each module consists of

a convolutional layer and a pooling layer. These modules are often stacked up with one on top of another, or with a DNN on top of it, to form a deep model [212]. The convolutional layer shares many weights, and the pooling layer subsamples the output of the convolutional layer and reduces the data rate from the layer below. The weight sharing in the convolutional layer, together with appropriately chosen pooling schemes, endows the CNN with some "invariance" properties (e.g., translation invariance). It has been argued that such limited "invariance" or equi-variance is not adequate for complex pattern recognition tasks and more principled ways of handling a wider range of invariance may be needed [160]. Nevertheless, CNNs have been found highly effective and been commonly used in computer vision and image recognition [54, 55, 56, 57, 69, 198, 209, 212, 434]. More recently, with appropriate changes from the CNN designed for image analysis to that taking into account speech-specific properties, the CNN is also found effective for speech recognition [1, 2, 3, 81, 94, 312]. We will discuss such applications in more detail in Section 7 of this monograph.

It is useful to point out that the time-delay neural network (TDNN) [202, 382] developed for early speech recognition is a special case and predecessor of the CNN when weight sharing is limited to one of the two dimensions, i.e., time dimension, and there is no pooling layer. It was not until recently that researchers have discovered that the time-dimension invariance is less important than the frequency-dimension invariance for speech recognition [1, 3, 81]. A careful analysis on the underlying reasons is described in [81], together with a new strategy for designing the CNN's pooling layer demonstrated to be more effective than all previous CNNs in phone recognition.

It is also useful to point out that the model of hierarchical temporal memory (HTM) [126, 143, 142] is another variant and extension of the CNN. The extension includes the following aspects: (1) Time or temporal dimension is introduced to serve as the "supervision" information for discrimination (even for static images); (2) Both bottom-up and top-down information flows are used, instead of just bottom-up in the CNN; and (3) A Bayesian probabilistic formalism is used for fusing information and for decision making.

Finally, the learning architecture developed for bottom-up, detection-based speech recognition proposed in [214] and developed further since 2004, notably in [330, 332, 427] using the DBN–DNN technique, can also be categorized in the discriminative or supervised-learning deep architecture category. There is no intent and mechanism in this architecture to characterize the joint probability of data and recognition targets of speech attributes and of the higher-level phone and words. The most current implementation of this approach is based on the DNN, or neural networks with many layers using back-propagation learning. One intermediate neural network layer in the implementation of this detection-based framework explicitly represents the speech attributes, which are simplified entities from the "atomic" units of speech developed in the early work of [101, 355]. The simplification lies in the removal of the temporally overlapping properties of the speech attributes or articulatory-like features. Embedding such more realistic properties in the future work is expected to improve the accuracy of speech recognition further.

3.4 Hybrid deep networks

The term "hybrid" for this third category refers to the deep architecture that either comprises or makes use of both generative and discriminative model components. In the existing hybrid architectures published in the literature, the generative component is mostly exploited to help with discrimination, which is the final goal of the hybrid architecture. How and why generative modeling can help with discrimination can be examined from two viewpoints [114]:

- The optimization viewpoint where generative models trained in an unsupervised fashion can provide excellent initialization points in highly nonlinear parameter estimation problems (The commonly used term of "pre-training" in deep learning has been introduced for this reason); and/or
- The regularization perspective where the unsupervised-learning models can effectively provide a prior on the set of functions representable by the model.

The study reported in [114] provided an insightful analysis and experimental evidence supporting both of the viewpoints above.

The DBN, a generative, deep network for unsupervised learning discussed in Section 3.2, can be converted to and used as the initial model of a DNN for supervised learning with the same network structure, which is further discriminatively trained or fine-tuned using the target labels provided. When the DBN is used in this way we consider this DBN–DNN model as a hybrid deep model, where the model trained using unsupervised data helps to make the discriminative model effective for supervised learning. We will review details of the discriminative DNN for supervised learning in the context of RBM/DBN generative, unsupervised pre-training in Section 5.

Another example of the hybrid deep network is developed in [260], where the DNN weights are also initialized from a generative DBN but are further fine-tuned with a sequence-level discriminative criterion, which is the conditional probability of the label sequence given the input feature sequence, instead of the frame-level criterion of cross-entropy commonly used. This can be viewed as a combination of the static DNN with the shallow discriminative architecture of CRF. It can be shown that such a DNN–CRF is equivalent to a hybrid deep architecture of DNN and HMM whose parameters are learned jointly using the full-sequence maximum mutual information (MMI) criterion between the entire label sequence and the input feature sequence. A closely related full-sequence training method designed and implemented for much larger tasks is carried out more recently with success for a shallow neural network [194] and for a deep one [195, 353, 374]. We note that the origin of the idea for joint training of the sequence model (e.g., the HMM) and of the neural network came from the early work of [17, 25], where shallow neural networks were trained with small amounts of training data and with no generative pre-training.

Here, it is useful to point out a connection between the above pretraining/fine-tuning strategy associated with hybrid deep networks and the highly popular minimum phone error (MPE) training technique for the HMM (see [147, 290] for an overview). To make MPE training effective, the parameters need to be initialized using an algorithm (e.g., Baum-Welch algorithm) that optimizes a generative criterion (e.g.,

maximum likelihood). This type of methods, which uses maximum-likelihood trained parameters to assist in the discriminative HMM training can be viewed as a "hybrid" approach to train the shallow HMM model.

Along the line of using discriminative criteria to train parameters in generative models as in the above HMM training example, we here discuss the same method applied to learning other hybrid deep networks. In [203], the generative model of RBM is learned using the discriminative criterion of posterior class-label probabilities. Here the label vector is concatenated with the input data vector to form the combined visible layer in the RBM. In this way, RBM can serve as a stand-alone solution to classification problems and the authors derived a discriminative learning algorithm for RBM as a shallow generative model. In the more recent work by Ranzato et al. [298], the deep generative model of DBN with gated Markov random field (MRF) at the lowest level is learned for feature extraction and then for recognition of difficult image classes including occlusions. The generative ability of the DBN facilitates the discovery of what information is captured and what is lost at each level of representation in the deep model, as demonstrated in [298]. A related study on using the discriminative criterion of empirical risk to train deep graphical models can be found in [352].

A further example of hybrid deep networks is the use of generative models of DBNs to pre-train deep convolutional neural networks (deep CNNs) [215, 216, 217]. Like the fully connected DNN discussed earlier, pre-training also helps to improve the performance of deep CNNs over random initialization. Pre-training DNNs or CNNs using a set of regularized deep autoencoders [24], including denoising autoencoders, contractive autoencoders, and sparse autoencoders, is also a similar example of the category of hybrid deep networks.

The final example given here for hybrid deep networks is based on the idea and work of [144, 267], where one task of discrimination (e.g., speech recognition) produces the output (text) that serves as the input to the second task of discrimination (e.g., machine translation). The overall system, giving the functionality of speech translation — translating speech in one language into text in another language — is a two-stage deep architecture consisting of both

generative and discriminative elements. Both models of speech recognition (e.g., HMM) and of machine translation (e.g., phrasal mapping and non-monotonic alignment) are generative in nature, but their parameters are all learned for discrimination of the ultimate translated text given the speech data. The framework described in [144] enables end-to-end performance optimization in the overall deep architecture using the unified learning framework initially published in [147]. This hybrid deep learning approach can be applied to not only speech translation but also all speech-centric and possibly other information processing tasks such as speech information retrieval, speech understanding, cross-lingual speech/text understanding and retrieval, etc. (e.g., [88, 94, 145, 146, 366, 398]).

In the next three chapters, we will elaborate on three prominent types of models for deep learning, one from each of the three classes reviewed in this chapter. These are chosen to serve the tutorial purpose, given their simplicity of the architectural and mathematical descriptions. The three architectures described in the following three chapters may not be interpreted as the most representative and influential work in each of the three classes.