An Overview of Deep-Structured Learning for Information Processing

Li Deng
Microsoft Research, Redmond, WA 98052, USA
E-mail: deng@microsoft.com, Tel: 425-706-2719

***Abstract*— In this paper, I will introduce to the APSIPA audience an emerging area of machine learning, deep-structured learning. It refers to a class of machine learning techniques, developed mostly since 2006, where many layers of information processing stages in hierarchical architectures are exploited for pattern classification and for unsupervised feature learning. First, the brief history of deep learning is discussed. Then, I develop a classificatory scheme to analyze and summarize major work reported in the deep learning literature. Using this scheme, I provide a taxonomy-oriented survey on the existing deep architectures, and categorize them into three types: generative, discriminative, and hybrid. Two prime deep architectures, one hybrid and one discriminative, are presented in detail. Finally, selected applications of deep learning are reviewed in broad areas of information processing including audio/speech, image/video, multimodality, language modeling, natural language processing, and information retrieval.**

I. Introduction

Today, signal processing research has a significantly widened scope compared to just a few years ago, and has encompassed many broad areas of information processing (Deng, 2008). In particular, machine learning has become an important technical area of the IEEE signal processing society. Since 2006, deep structured learning, or more commonly called deep learning, has emerged as a new area of machine learning research (Hinton et al., 2006). Within the past few years, the techniques developed from deep learning research has already been impacting a wide range of signal and information processing work within the traditional and the new, widened scopes (Yu and Deng, 2011). A series of workshops, such as the 2011 ICML Workshop on Learning Architectures, Representations, and Optimization for Speech and Visual Information Processing, the 2011 Learning Workshop, the 2009 ICML Workshop on Learning Feature Hierarchies, the 2009 NIPS Workshop on Deep Learning for Speech Recognition and Related Applications, the 2008 NIPS Deep Learning Workshop, as well as the special section on Deep Learning for Speech and Language Processing in IEEE Transactions on Audio, Speech, and Language Processing (to appear in January 2012) have been devoted exclusively to deep learning and its applications to various classical and expanded signal processing areas. We have also seen the government sponsored research on deep learning; e.g., the DARPA deep learning program (DARPA, 2009). The author has been directly involved in organizing several of the events above, and has seen the emerging nature of deep learning; Hence the need for providing an overview article here.

Deep learning refers to a class of machine learning techniques, where many layers of information processing stages in hierarchical architectures are exploited for pattern classification and for unsupervised feature learning. It is in the intersection among the research areas of neural network, graphical modeling, 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).

This overview paper, as the companion material to the invited tutorial, is aimed to introduce the tutorial audience and the readers of this article to the emerging technologies enabled by deep learning. I also attempt to provide a comprehensive review on the research work conducted in this exciting area since the birth of deep learning in 2006 which is of direct relevance to signal and information processing. Future research directions will be discussed to attract interests of and to solicit efforts from more signal processing researchers, students, and practitioners in this emerging area for advancing signal and information processing technology and applications.

II. Deep-Structured Learning --- A Brief Historical Account

Until recently, most machine learning and signal processing techniques had exploited shallow-structured architectures. These architectures typically contain a single layer of nonlinear feature transformations and they lack multiple layers of adaptive non-linear features. Examples of the shallow architectures are conventional hidden Markov models (HMMs), linear or nonlinear dynamical systems, conditional random fields (CRFs), maximum entropy (MaxEnt) models, support vector machines (SVMs), logistic regression, kernel regression, and multi-layer perceptron (MLP) neural network with a single hidden layer. A property common to these shallow learning models is the simple architecture that consists of only one layer responsible for transforming the raw input signals or features into a problem-specific feature space, which may be unobservable. Take the example of a SVM. It is a shallow linear separation model with one or zero feature transformation layer when kernel trick is and is not used, respectively. Shallow architectures have been shown effective in solving many simple or well-constrained problems, but their limited modeling and representational power can cause difficulties when dealing with more complicated real-world applications involving natural signals such as human speech, natural sound and language, and natural image and visual scenes.

Human information processing mechanisms (e.g., vision and speech), however, suggest the need of deep architectures for extracting complex structure and building internal representation from rich sensory inputs. For example, human speech production and perception systems are both equipped with clearly layered hierarchical structures in transforming the information from the waveform level to the linguistic level (Baker et al., 2009; Deng, 1999, 2003). It is natural to believe that the state-of-the-art can be advanced in processing these types of natural signals if efficient and effective deep learning algorithms are developed. Signal processing systems with deep architectures are composed of many layers of nonlinear processing stages, where each lower layer's outputs are fed to its immediate higher layer as the input. The successful deep learning techniques developed so far share two additional key properties: the generative nature of the model, which typically requires adding an additional top layer to perform the discriminative task, and an *unsupervised* pre-training step that makes effective use of large amounts of unlabeled training data for extracting structures and regularities in the input features.

Historically, the concept of deep learning was originated from artificial neural network research. (Hence, one may occasionally hear the discussion of "the third generation neural networks".) MLP neural networks with many hidden layers are indeed a good example of the models with deep architectures. Back-propagation, invented in 1980's, has been a well-known algorithm for learning the weights of these networks. Unfortunately back-propagation alone does not work well in practice for learning networks with more than a small number of hidden layers (see a review and interesting analysis in (Bengio, 2009; Glorot and Bengio, 2010). The pervasive presence of local optima in the non-convex objective function of the deep networks is the main source of difficulty in the learning. Back-propagation is based on local gradient descent, and starts usually at some random initial points. It often gets trapped in poor local optima, and the severity increases significantly as the depth of the networks increases. This difficulty is partially responsible for steering away most of the machine learning and signal processing research from neural networks to shallow models that have convex loss functions (e.g., SVMs, CRFs, and MaxEnt models) for which global optimum can be efficiently obtained at the cost of less powerful models.

The optimization difficulty associated with the deep models was empirically alleviated when a reasonably efficient, unsupervised learning algorithm was introduced in the two papers of (Hinton et al. 2006; Hinton and Salakhutdinov, 2006). In these papers, a class of deep generative models was introduced, called deep belief networks (DBNs). A core component of the DBN is a greedy, layer-by-layer learning algorithm which optimizes DBN weights at time complexity linear to the size and depth of the networks. Separately and with some surprise, initializing the weights of an MLP with a correspondingly configured DBN often produces much better results than that with the random weights. As such, deep networks that are learned with unsupervised DBN pre-training followed by the back-propagation fine-tuning is also called DBNs in the literature (e.g., Dahl et al., 2012; Mohamed et al., 2010, 2012).

In addition to the supply of good initialization points, DBN comes with additional attractive features. First, the learning algorithm makes effective use of unlabeled data. Second, it can be interpreted as Bayesian probabilistic generative model. Third, the values of the hidden variables in the deepest layer are efficient to compute. And fourth, the over-fitting problem, which is often observed in the models with millions of parameters such as DBNs, and the under-fitting problem, which occurs often in deep networks, are effectively addressed by the generative pre-training step.

The DBN training procedure is not the only one that makes deep learning possible. Since the publication of the seminal work in (Hinton et al., 2006; Hinton and Salakhutdinov, 2006), a number of other researchers have been improving and applying the deep learning techniques with success. For example, one can alternatively pre-train the deep networks layer by layer by considering each pair of layers as a de-noising auto-encoder (Bengio, 2009).

In the next section, a brief overview is provided on the various architectures of deep learning, including and beyond the original DBN.

III. Deep Learning Architectures --- A Taxonomy-Oriented Overview

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 stages 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 categorize most of the work in this area into three types:

- Generative deep architectures, which are intended to characterize the high-order correlation properties of the data or joint statistical distributions of the visible data and their associated classes. Use of Bayes rule can turn this type of architecture into a discriminative one.
- Discriminative deep architectures, which are intended to provide discriminative power for pattern classification,

often by characterizing the posterior distributions of classes conditioned on the visible data; and

- Hybrid deep architectures, where the goal is discrimination but is assisted (often in a significant way) with the outcomes of generative architectures via better optimization or/and regularization.

Below we briefly describe the representative work in each of the above three categories.

A. Generative architecture

Associated with this generative category, we often see "unsupervised learning", since the labels for the data are not of concern. Among the various subclasses of generative deep architecture, the energy-based deep models are the most common (e.g., Ngiam et al., 2011; Bengio, 2009; LeCun et al., 2007). One typical case is stacked de-noising auto-encoder (Vincent et al., 2010). Other forms of deep auto-encoders are also generative in nature, but with quite different properties and implementations. Examples are transforming auto-encoders (Hinton et al., 2010) and the original form of the auto-encoders (Hinton and Salakhutdinov, 2006; Deng et al., 2010) implemented by stacked RBMs.

Another prominent type of energy-based generative model is deep Boltzmann machine or DBM (Salakhutdinov and Hinton, 2009). 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 make stochastic decisions about whether to be on or off. While having very simple learning algorithm, the general BMs are very complex to study and very slow to compute in learning. 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, but RBM is much faster to learn. 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 IV.

The standard DBN has been extended to the factored higher-order Boltzmann machine in its bottom layer, with strong results for phone recognition (Dahl et. al., 2010). But it is very difficult to train this layer and the results are not easy to reproduce. Other representative deep generative architectures include sum-product networks (Poon and Domingos et al., 2011), and recurrent neural networks, which are demonstrated to be well capable of generating sequential data such as text characters (Sutskever et al., 2011).

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 book (Deng, 2006). Specifically, the early work described in (Deng 1992, 1993; Deng et al., 1994; Ostendorf et al., 1996, Deng and Sameti, 1996) generalized and extended the conventional shallow 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 with superior learning techniques for time-varying HMM parameters and with the applications extended to speech recognition robustness; e.g., Yu and Deng, 2009; Yu et al., 2009. See also Zen et al., 2011 for the use of trajectory HMMs for feature mapping.). Subsequent work added a new hidden layer into the dynamic model so as to explicitly account for the target-directed, articulatory-like properties in human speech generation (Deng and Ramsay, 1997; Deng, 1998; Bridle et al., 1998; Deng, 1999; Picone et al., 1999; Deng, 2003). More efficient implementation of this deep architecture with hidden dynamics is achieved with non-recursive or FIR filters in more recent studies (Deng et. al., 2006a. 2006b, Deng and Yu, 2007). All the above deep-structured generative models of speech can be shown as special cases of the more general dynamic Bayesian network model and even more general dynamic graphical models (Bilmes and Bartels, 2005; Bilmes, 2010). 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 (Rennie et al., 2010; Wohlmayr et al., 2011).

Dynamic or temporally recursive generative models for non-speech applications based on the deep neural network architecture can be found in (Taylor et al., 2007) for human motion modeling, and in (Socher et al., 2011) for natural language parsing.

B. Discriminative architecture

Many of the discriminative techniques in signal and information processing apply to shallow architectures such as HMMs (e.g., He et al., 2008; Jiang and Li, 2010; Xiao and Deng, 2010; Gibson and Hain, 2010) or CRFs (e.g., Yang and

Furui, 2009; Yu, Li, and Deng, 2010; Hifny and Renals, 2009). Since CRF is defined with the conditional probability on data, it is intrinsically a (shallow) discriminative architecture. 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 (Yu, Wang, and Deng, 2010). Various versions of deep-structured CRFs are usefully applied to phone recognition (Yu and Deng, 2010), spoken language identification (Yu, Wang, Karam, and Deng, 2010), and natural language processing (Yu, Wang, and Deng, 2010). However, at least for the phone recognition task, the performance of deep-structured CRFs, which is purely discriminative and non-generative, has not been able to match that of the hybrid approach involving DBN, which we will take on shortly.

The recent article of (Morgan, 2012) gives an excellent review on other major existing discriminative deep 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, where a discriminatively learned neural network is developed in the context of computing discriminant emission probabilities for HMMs; for the most recent work, see (Pinto et al., 2011; Ketabdar and Bourlard, 2010). The tandem approach generates features for the HMM for phonetic classification, using one or more hidden layers of neural network with various ways of information combination (Morgan et al., 2005; Morgan 2011).

In the most recent work of (Deng et. al, 2011), a new deep learning architecture, called Deep Convex Network or DCN, is developed which focuses on discrimination with scalable, parallelizable learning and has no generative component. We will describe this discriminative deep architecture in detail in Section V.

Finally, the learning architecture developed for bottom-up, detection-based speech recognition proposed in (Lee, 2004) and developed since then can also be categorized in this discriminative deep architecture category. There is no intent and mechanism in the model to characterize joint probability of data and the recognition targets of speech attributes and the subsequent phone and words. The most current implementation is based on multiple layers of neural network 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 (Deng and Sun, 1994). The simplification lies in the removal of the temporally overlapping properties of the speech attributes or articulatory-like features. Embedding such properties is expected to improve the accuracy of speech recognition.

C. Hybrid architecture

Hybrid deep architecture in this category refers to the deep architecture that comprises and makes use of both generative and discriminative components. In the existing hybrid architectures published in the current literature, the generative component is mostly exploited to help with discrimination as the final goal of the hybrid architecture. How and why generative modeling can help with discriminative can be examined from two viewpoints:

1) The optimization viewpoint where generative models 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

2) The regularization perspective where generative models can effectively control the complexity of the overall model. See (Erhan et al., 2010) for an insightful analysis on and experimental evidence supporting both of the viewpoints above.

When the generative deep architecture of DBN discussed in Subsection III.A is subject to further discriminative training, commonly called “fine-tuning” in the literature, we obtain an equivalent architecture of deep neural network (DNN, which is also called DBN or deep MLP in the literature). In the DNN or the hybrid DBN with fine tuning, the weights of the network are “pre-trained” from RBM and DBN instead of the usual random initialization. See (Mohamed et al, 2012) for a detailed explanation of the equivalence relationship and the use of the often confusing terminology. We will review details of the DNN in the context of RBM/DBN pre-training and its interface with the most commonly used shallow generative architecture of HMM (DNN-HMM or DBN-HMM) in Section IV.

Another typical example of the hybrid deep architecture is developed in (Mohamed et al., 2010). This is a hybrid of DNN with a shallow discriminative architecture of conditional random field (CRF). Here, the overall architecture of DNN-CRF is learned using the discriminative criterion of frame-level conditional probability of labels given the input data. It can be shown that such 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) between the entire label sequence and the input vector sequence.

A final example given here of the hybrid deep architecture is based on the work of (He and Deng, 2011), where one task of discrimination (speech recognition) produces the output (text) that serves as the input to the second task of discrimination (machine translation). The overall system, giving the functionality of speech translation --- translating speech in one language into text in another language --- is a 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) are generative in nature. But their parameters are all learned for discrimination. The framework described in (He and Deng, 2011) enables end-to-end performance optimization in the overall deep architecture using the unified learning framework initially published in (He et al., 2008). 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.

In the following two sections, I will elaborate on two example architectures of deep learning.

IV. Hybrid Architecture: Deep Belief Network

A. Basics

In this section, we present the most widely studied hybrid deep architecture of DBN or DNN, consisting of both pre-training and fine-tuning stages in its parameter learning. Part of this review is based on the recent publication of (Yu and Deng, 2011).

As the generative component of the DBN, it is a probabilistic model composed of multiple layers of stochastic, latent variables. The unobserved variables can have binary values and are often called hidden units or feature detectors. The top two layers have undirected, symmetric connections between them and form an *associative memory*. The lower layers receive top-down, directed connections from the layer above. The states of the units in the lowest layer, or the visible units, represent an input data vector.

There is an efficient, layer-by-layer procedure for learning the top-down, generative weights that determine how the variables in one layer depend on the variables in the layer above. After learning, the values of the latent variables in every layer can be inferred by a single, bottom-up pass that starts with an observed data vector in the bottom layer and uses the generative weights in the reverse direction.

DBNs are learned one layer at a time by treating the values of the latent variables in one layer, when they are being inferred from data, as the data for training the next layer. This efficient, greedy learning can be followed by, or combined with, other learning procedures that fine-tune all of the weights to improve the generative or discriminative performance of the full network. This latter learning procedure constitutes the discriminative component of the DBN as the hybrid architecture.

Discriminative fine-tuning can be performed by adding a final layer of variables that represent the desired outputs and back-propagating error derivatives. When networks with many hidden layers are applied to highly-structured input data, such as speech and images, back-propagation works much better if the feature detectors in the hidden layers are initialized by learning a DBN to model the structure in the input data as originally proposed in (Hinton and Salakhutdinov, 2006).

A DBN can be viewed as a composition of simple learning modules via stacking them. This simple learning module is called restricted Boltzmann machines (RBMs) that we introduce next.

B. Restricted Boltzmann Machine

An RBM is a special type of Markov random field that has one layer of (typically Bernoulli) stochastic hidden units and one layer of (typically Bernoulli or Gaussian) stochastic visible or observable units. RBMs can be represented as bipartite graphs, where all visible units are connected to all hidden units, and there are no visible-visible or hidden-hidden connections.

In an RBM, the joint distribution $\mathrm{p}(\mathbf{v},\mathbf{h};\theta)$ over the visible units $\mathbf{v}$ and hidden units $\mathbf{h}$, given the model parameters θ, is defined in terms of an energy function $\mathrm{E}(\mathbf{v},\mathbf{h};\theta)$ of

$$\mathrm{p}(\mathbf{v},\mathbf{h};\theta) = \frac{exp\big(-\mathrm{E}(\mathbf{v},\mathbf{h};\theta)\big)}{Z},$$

where $Z = \sum_{\mathbf{v}}\sum_{\mathbf{h}} exp\big(-\mathrm{E}(\mathbf{v},\mathbf{h};\theta)\big)$ is a normalization factor or partition function, and the marginal probability that the model assigns to a visible vector $\mathbf{v}$ is

$$\mathrm{p}(\mathbf{v};\theta) = \frac{\sum_{\boldsymbol{h}} exp\big(-\mathrm{E}(\mathbf{v},\mathbf{h};\theta)\big)}{Z}.$$

For a Bernoulli (visible)-Bernoulli (hidden) RBM, the energy function is defined as

$$\mathrm{E}(\mathbf{v},\mathbf{h};\theta) = -\sum_{i=1}^{I}\sum_{j=1}^{J} w_{ij}\, v_i h_j - \sum_{i=1}^{I} b_i v_i - \sum_{j=1}^{J} a_j h_j,$$

where w_{ij} represents the symmetric interaction term between visible unit v_i and hidden unit h_j, b_i and a_j the bias terms, and I and J are the numbers of visible and hidden units. The conditional probabilities can be efficiently calculated as

$$p\big(h_j = 1|\mathbf{v};\theta\big) = \sigma\left(\sum_{i=1}^{I} w_{ij}\, v_i + a_j\right),$$

$$p(v_i = 1|\mathbf{h};\theta) = \sigma\left(\sum_{j=1}^{J} w_{ij}\, h_j + b_i\right),$$

where $\sigma(x) = 1/\big(1 + exp(x)\big)$.

Similarly, for a Gaussian (visible)-Bernoulli (hidden) RBM, the energy is

$$E(\mathbf{v},\mathbf{h};\theta) = -\sum_{i=1}^{I}\sum_{j=1}^{J} w_{ij}\, v_i h_j - \frac{1}{2}\sum_{i=1}^{I}(v_i - b_i)^2 - \sum_{j=1}^{J} a_j h_j,$$

The corresponding conditional probabilities become

$$p(h_j = 1|\mathbf{v};\theta) = \sigma\left(\sum_{i=1}^{I} w_{ij}\, v_i + a_j\right),$$

$$p(v_i|\mathbf{h};\theta) = \mathcal{N}\left(\sum_{j=1}^{J} w_{ij}\, h_j + b_i, 1\right),$$

where v_i takes real values and follows a Gaussian distribution with mean $\sum_{j=1}^{J} w_{ij}\, h_j + b_i$ and variance one. Gaussian-Bernoulli RBMs can be used to convert real-valued stochastic variables to binary stochastic variables, which can then be further processed using the Bernoulli-Bernoulli RBMs.

Taking the gradient of the log likelihood $\log p(\mathbf{v};\theta)$ we can derive the update rule for the RBM weights as:

$$\Delta w_{ij} = E_{data}(v_i h_j) - E_{model}(v_i h_j),$$

where $E_{data}(v_i h_j)$ is the expectation observed in the training set and $E_{model}(v_i h_j)$ is that same expectation under the distribution defined by the model. Unfortunately, $E_{model}(v_i h_j)$ is intractable to compute so the contrastive divergence (CD) approximation to the gradient is used where $E_{model}(v_i h_j)$ is replaced by running the Gibbs sampler initialized at the data for one full step. Careful training of RBMs is essential to the success of applying deep learning to practical problems. A practical guide of the RBM training is provided in (Hinton, 2010).

C. Stacking up RBM to form a DBN

Stacking a number of the RBMs learned layer by layer from bottom up gives rise to a DBN an example of which is shown in Fig. 1. The stacking procedure is as follows. After learning a Gaussian-Bernoulli RBM (for applications with continuous features such as speech) or Bernoulli-Bernoulli RBM (for applications with nominal or binary features such as black-white image or coded text), we treat the activation probabilities of its hidden units as the data for training the Bernoulli-Bernoulli RBM one layer up. The activation probabilities of the second-layer Bernoulli-Bernoulli RBM are then used as the visible data input for the third-layer Bernoulli-Bernoulli RBM, and so on. Theoretical justification of this efficient layer-by-layer greedy learning strategy is given in (Hinton et al., 2006), where it is shown that the *stacking* procedure above improves a variational lower bound on the likelihood of the training data under the composite model. That is, the greedy procedure above achieves approximate maximum likelihood learning. Note that this learning procedure is unsupervised and requires no class label.

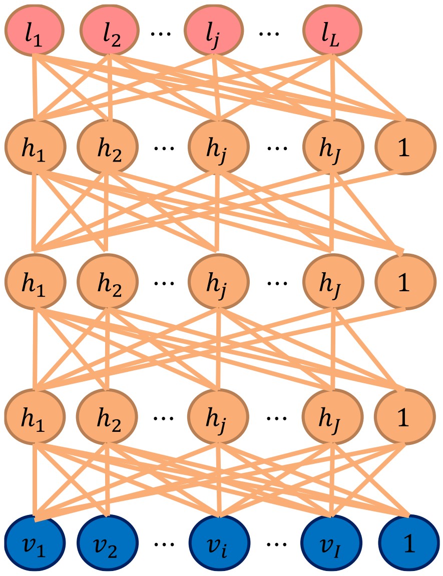

l_1 l_2 ⋯ l_j ⋯ l_L
h_1 h_2 ⋯ h_j ⋯ h_J 1
h_1 h_2 ⋯ h_j ⋯ h_J 1
h_1 h_2 ⋯ h_j ⋯ h_J 1
v_1 v_2 ⋯ v_i ⋯ v_I 1

Fig. 1: Illustration of a DBN architecture.

When DBN is applied to classification tasks, the generative pre-training can be followed by or combined with other, typically discriminative, learning procedures that fine-tune all of the weights jointly to improve the performance of the DBN. This discriminative fine-tuning is often performed by adding a final layer of variables that represent the desired outputs or labels provided in the training data. Then, the back-propagation algorithm can be used to adjust or fine-tune the DBN weights. For example, for speech recognition the output layer can represent either syllables, phones, sub-phones, phone states, or other speech units used in the HMM-based speech recognition system.

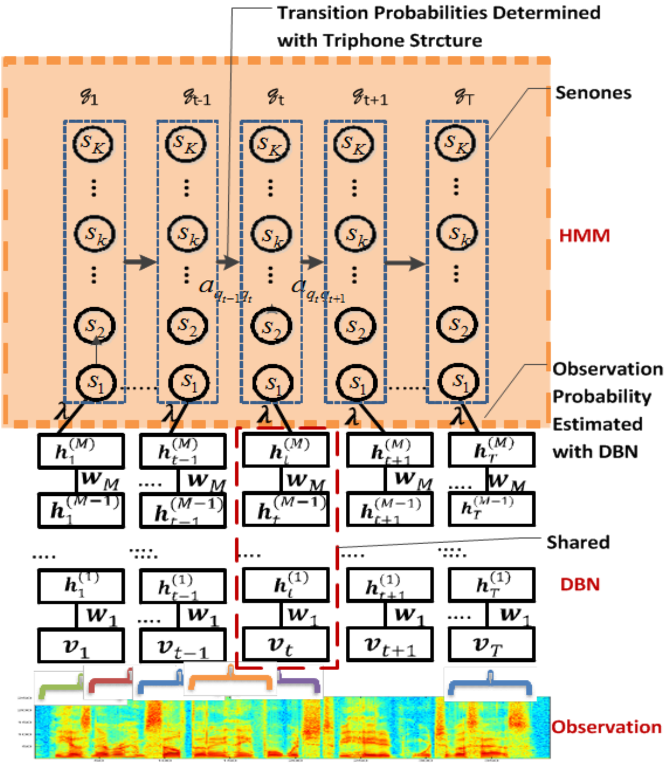
D. Interfacing DBN with HMM

A DBN is a static classifier with input vectors having a fixed dimensionality. However, many practical pattern recognition and information processing problems, including speech recognition, machine translation, natural language understanding, video processing and bio-information processing, require sequence recognition. In sequence recognition, sometimes called classification with structured input/output, the dimensionality of both inputs and outputs are variable.

The HMM, based on dynamic programing operations, is a convenient tool to help port the strength of a static classifier to handle dynamic or sequential patterns. Thus, it is natural to combine DBN and HMM to bridge the gap between static and sequence pattern recognition. An architecture that shows the interface between a DBN and HMM is provided in Fig. 2.

This architecture has been successfully used in speech recognition experiments reported in (Dahl, Yu, Deng, and Acero, 2012).

It is important to note that the unique elasticity of temporal dynamic of speech as elaborated in the book (Deng, 2006) would require temporally-correlated models better than HMM for the ultimate success of speech recognition. Integrating such dynamic models having realistic co-articulatory properties with the DBN and possibly other deep learning models to form the coherent dynamic deep architecture is a very challenging new research.



Transition Probabilities Determined with Triphone Strcture
Senones
HMM
Observation Probability Estimated with DBN
Shared
DBN
Observation

Fig. 2 Interface between DBN and HMM to form a DBN-HMM or DNN-HMM. This architecture has been successfully used in speech recognition experiments reported in (Dahl, Yu, Deng, and Acero, 2012).

V. Discriminative Architecture: Deep Convex Network

A. *Motivations*

While DBN just reviewed has been shown to be extremely powerful in connection with performing recognition and classification tasks including speech recognition and image classification, training DBN has proven to be more difficult computationally. In particular, conventional techniques for training DBN at the fine tuning phase involve the utilization of a stochastic gradient descent learning algorithm, which is extremely difficult to parallelize across machines. This makes learning at large scale practically impossible. For example, it has been possible to use one single, very powerful GPU machine to train DBN-based speech recognizers with dozens to a few hundreds of hours of speech training data with remarkable results. It is very difficult, however, to scale up this success with thousands or more hours of training data.

Here we describe a new deep learning architecture, Deep Convex Network (DCN), which attacks the learning scalability problem. This section is based in part on the recent publication of (Deng and Yu, 2011).

B. *An architectural overview of DCN*

A DCN, shown in Fig. 3, includes a variable number of layered modules, wherein each module is a specialized neural network consisting of a single hidden layer and two trainable sets of weights. More particularly, the lowest module in the DCN comprises a first linear layer with a set of linear input units, a non-linear layer with a set of non-linear hidden units, and a second linear layer with a set of linear output units. For instance, if the DCN is utilized in connection with recognizing an image, the input units can correspond to a number of pixels (or extracted features) in the image, and can be assigned values based at least in part upon intensity values, RGB values, or the like corresponding to the respective pixels. If the DCN is utilized in connection with speech recognition, the set of input units may correspond to samples of speech waveform, or the extracted features from speech waveforms, such as power spectra or cepstral coefficients.

The hidden layer of the *lowest module* of a DCN comprises a set of non-linear units that are mapped to the input units by way of a first, lower-layer weight matrix, which we denote by ***W***. For instance, the weight matrix may comprise a plurality of randomly generated values between zero and one, or the weights of an RBM trained separately. The non-linear units may be sigmoidal units that are configured to perform non-linear operations on weighted outputs from the input units (weighted in accordance with the first weight matrix ***W***).

The second, *linear* layer in any module of a DCN includes a set of output units that are representative of the targets of classification. For instance, if the DCN is configured to perform digit recognition, then the plurality of output units may be representative of the values 1, 2, 3, and so forth up to 10 with a 0-1 coding scheme. If the DCN is configured to perform speech recognition, then the output units may be representative of phones, HMM states of phones, or context-dependent HMM states of phones. The non-linear units in each module of the DCN may be mapped to a set of the linear output units by way of a second, upper-layer weight matrix, which we denote by ***U***. This second weight matrix can be learned by way of a batch learning process, such that learning can be undertaken in parallel. Convex optimization can be employed in connection with learning ***U***. For instance, ***U*** can be learned based at least in part upon the first weight matrix

***W*,** values of the coded classification targets, and values of the input units.

As indicated above, the DCN includes a set of serially connected, overlapping, and layered modules, wherein each module includes the aforementioned three layers -- a first linear layer that includes a set of linear input units whose number equals the dimensionality of the input features, a hidden layer that comprises a set of non-linear units whose number is a tunable hyper-parameter, and a second linear layer that comprises a plurality of linear output units whose number equals that of the target classification classes (e.g., the total number of context-dependent phones clustered by a decision tree used in). The modules are referred to herein as being layered because the output units of a lower module are a subset of the input units of an adjacent higher module in the DCN. More specifically, in a second module that is directly above the lowest module in the DCN, the input units can include the output units of the lower module(s). The input units can additionally include the raw training data – in other words, the output units of the lowest module can be appended to the input units in the second module, such that the input units of the second module also include the output units of the lowest module.

The pattern discussed above of including output units in a lower module as a portion of the input units in an adjacent higher module in the DBN and thereafter learning a weight matrix that describes connection weights between hidden units and linear output units via convex optimization can continue for many modules. A resultant learned DCN may then be deployed in connection with an automatic classification task such as frame-level speech phone or state classification. Connecting DCN's output to an HMM or any dynamic programming device enables continuous speech recognition and other forms of sequential pattern recognition.

VI. Applications of Deep Learning to Signal and Information Processing

In the expanded technical scope of signal processing, the *signal* is endowed with not only the traditional types such as audio, speech, image and video, but also text, language, and document that convey high-level, semantic information for human consumption. In addition, the scope of *processing* has been extended from the conventional coding, enhancement, analysis, and recognition to include more human-centric tasks of interpretation, understanding, retrieval, mining, and user interface (Deng, 2008). Many signal processing researchers have been working on one or more of the signal processing areas defined by the matrix constructed with the two axes of *signal* and *processing* discussed here. The deep learning techniques discussed in this article have recently been applied to quite a number of extended signal processing areas. We now provide a brief survey of this body of work in four main categories.

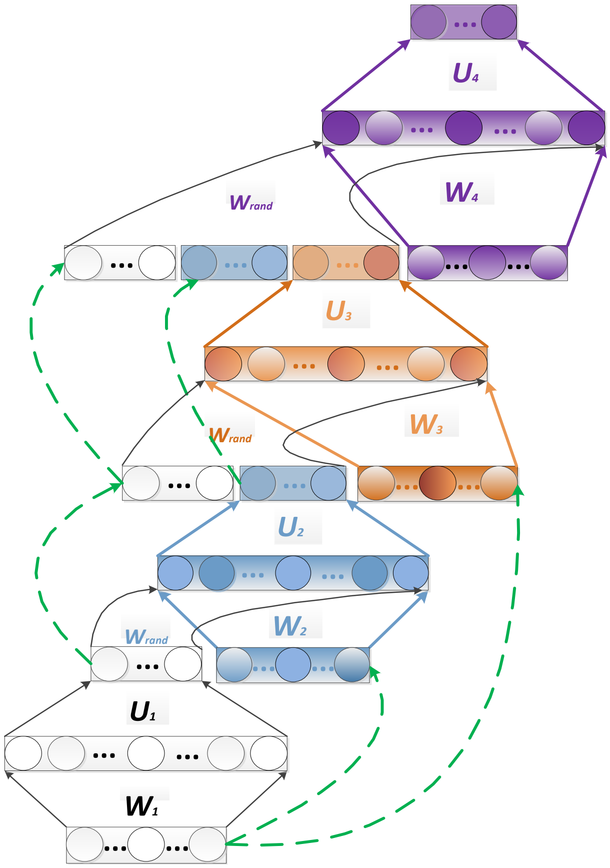


Fig. 3: A DCN architecture. Only four modules are illustrated. In practice, up to a few hundreds of modules have been efficiently trained and used in image and speech classification experiments.

A. *Speech and audio*

The traditional neural network or MLP has been in use for speech recognition for many years. When used alone, its performance is typically lower than the state-of-the-art HMM systems with observation probabilities approximated with Gaussian mixture models (GMMs). Recently, the deep learning technique was successfully applied to phone recognition (Mohamed et al., 2009, 2010, 2012; Sivaram and Hermansky, 2012) and large vocabulary speech recognition tasks (Yu et al., 2012) by integrating the powerful discriminative training ability of the DBNs and the sequential modeling ability of the HMMs.

More specifically, the work of (Mohamed et al., 2009), a five-layer DBN was used to replace the Gaussian mixture component of the GMM-HMM and the monophone state was used as the modeling unit. Although monophones are generally accepted as a weaker phonetic representation than triphones, the DBN-HMM approach with monophones was shown to achieve higher phone recognition accuracy than the state-of-the-art triphone GMM-HMM systems.

The technique of (Mohamed et al., 2009) was improved in the later work reported in (Mohamed et al., 2010) by using the CRF instead of the HMM to model the sequential speech data and by applying the maximum mutual information (MMI) training technique successfully developed in speech recognition to the resultant DBN-CRF training. The sequential discriminative learning technique developed jointly optimizes the DBN weights, transition weights, and phone language model and achieved higher accuracy than the DBN-HMM phone recognizer with the frame-discriminative training criterion implicit in the DBN's fine tuning procedure implemented as implemented in (Mohamed et al., 2009).

In (Dahl et al., 2011, 2012), the DBN-HMM was extended from the monophone phonetic representation to the triphone or context-dependent counterpart and from phone recognition to large vocabulary speech recognition. Experiments on the Bing mobile voice search dataset collected under the real usage scenario demonstrate that the triphone DBN-HMM significantly outperforms the state-of-the-art HMM system. Three factors contribute to the success: the use of triphones as the DBN modeling units, the use of the best available tri-phone GMM-HMM to generate the senone alignment, and the tuning of the transition probabilities. Experiments also indicate that the decoding time of a five-layer DBN-HMM is almost the same as that of the state-of-the-art triphone GMM-HMM.

In (Deng et al., 2010), a type of deep auto-encoder developed originally for image feature coding was explored on the speech feature coding problem. The goal is to extract "bottle-neck" speech features by compressing the high-resolution speech spectrogram data to a pre-defined number of bits with minimal reproduction error. DBN pre-training is found to be crucial for high coding efficiency. When the DBN pre-training is used, the deep auto-encoder is shown to significantly outperform a traditional vector quantization technique. If weights in the deep auto-encoder are randomly initialized the performance is substantially degraded. The architecture of the deep auto-encoder used in that work is shown in Fig. 4, with two layers of RBM following by fine-tuning with four layers of DNN.

Further, the most recent work of (Deng and Yu, 2011) makes use of the DCN architecture to perform frame-level phone classification. Higher accuracy than DBN is reported, especially after a "fine-tuning" technique developed in (Yu and Deng, 2011) is exploited. While the preliminary work reported in (Deng and Yu, 2011) has not developed parallel implementation of the basic learning algorithm in the DCN architecture, active research is currently underway to enable high scalability of learning DCN via parallelization.

In the work reported in (Lee et al., 2009) and some follow-up work, the convolutional structure is further imposed on DBN and is applied to audio and speech data for a number of tasks including music artist and genre classification, speaker identification, speaker gender classification, and phone classification, with strong results presented.

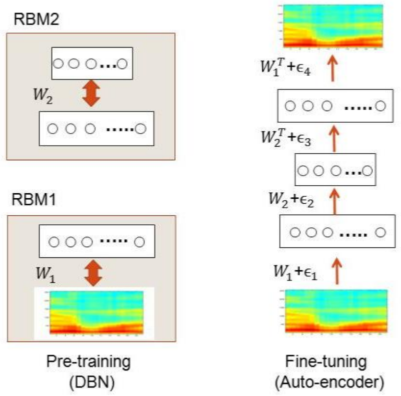


Fig. 4: The architecture of the deep auto-encoder used in (Deng et al., 2010) for extracting "bottle-neck" speech features from high-resolution spectrograms.

The recent work reported in (Jaitly and Hinton, 2011) makes use of speech sound waves as the raw input feature to an RBM with a convolutional structure as the classifier. With the use of rectifier linear units in the hidden layer (Glorot et al., 2011), it is possible to automatically normalize the amplitude variation in the waveform signal, thus overcoming the difficulty encountered in the earlier attempt of using the same raw feature in the HMM based approach (Sheikhzadeh and Deng, 1994).

In addition to RBM, DBN, and DCN, other deep models have also been developed and reported in the literature. For example, the deep-structured CRF, which stacks many layers of CRFs, have been successfully used in the task of language identification (Yu, Wang, Karam, and Deng, 2010), phone recognition (Yu and Deng, 2010)0, sequential labeling in natural language processing (Yu, Wang, and Deng, 2010), and confidence calibration in speech recognition (Yu, Li, and Deng, 2010).

As another example, in (Saon and Chien, 2012), a new type of HMM is introduced in which a set of hidden basis vectors and associated weights and precision matrices are jointly optimized. This can be considered as a generative deep architecture where the hidden basis, together with the associated weights, gives an intermediate representation of the speech signal. The work explores making HMM training more generalizable to unknown data, achieved by the developed Bayesian sensing framework to realize model regularization.

B. Image, video, and multimodality

The original DBN and deep auto-encoder were developed and demonstrated with success on the simple image recognition and dimensionality reduction (coding) tasks (MNIST) in (Hinton and Salakhutdinov, 2006). It is interesting to note that the gain of coding efficiency using the DBN-based auto-encoder on the image data over the conventional method of principal component analysis as demonstrated in ((Hinton and Salakhutdinov, 2006) is very similar to the gain reported in (Deng et al., 2010) on the speech data over the traditional technique of vector quantization.

In (Nair and Hinton, 2009), a modified DBN is developed where the top-layer model uses a third-order Boltzmann machine. This type of DBN is applied to the NORB database – a 3-dimensional object recognition task. An error rate close to the best published result on this task is reported. In particular, it is shown that the DBN substantially outperforms shallow models such as SVMs.

In (Tang and Eliasmith, 2010), two strategies to improve the robustness of the DBN are developed. First, sparse connections in the first layer of the DBN are used as a way to regularize the model. Second, a probabilistic denoising algorithm is developed. Both techniques are shown to be effective in improving the robustness against occlusion and random noise in a noisy image recognition task.

DBNs have also been successfully applied to create compact but meaningful representations of images (Taralba et al., 2008) for retrieval purposes. On this large collection image retrieval task, deep learning approaches also produced strong results.

Use of conditional DBN for video sequence and human motion synthesis is reported in (Taylor et al., 2007). The conditional DBN makes the DBN weights associated with a fixed time window conditioned on the data from previous time steps. The computational tool offered in this type of temporal DBN may provide the opportunity to improve the DBN-HMMs towards efficient integration of temporal-centric human speech production mechanisms into DBN-based speech production model.

A very interesting piece of recent work appears in (Ngiam et al., 2011), where the authors from Stanford propose and evaluate a novel application of deep networks to learn features over both audio and video modalities. Cross modality feature learning is demonstrated --- better features for video can be learned if both audio and video information sources are available at feature learning time. The authors further show how to learn a shared audio and video representation, and evaluate it on a fixed task, where the classifier is trained with audio-only data but tested with video-only data and vice-versa. The work concludes that deep learning architectures are effective in learning multimodal features from unlabeled data and in improving single modality features through cross modality learning.

C. Language modeling

Research in language, document, and text processing has seen increasing popularity recently in the signal processing community, and has been designated as one of the main focus areas by the society's audio, speech, and language processing technical committee. There has been a long history (e.g., Bengio et al., 2000; Zamora et al., 2009) of using (shallow) neural networks in language modeling (LM) – an important component in speech recognition, machine translation, text information retrieval, and in natural language processing. Recently, deep neural networks have been attracting more and more attention in statistical language modeling.

An LM is a function that captures the salient statistical characteristics of the distribution of sequences of words in a natural language. It allows one to make probabilistic predictions of the next word given preceding ones. A neural network LM is one that exploits the neural network ability to learn distributed representations to reduce the impact of the curse of dimensionality.

A distributed representation of a symbol is a vector of features which characterize the meaning of the symbol. With a neural network LM, one relies on the learning algorithm to discover meaningful, continuous-valued features. The basic idea is to learn to associate each word in the dictionary with a continuous-valued vector representation, where each word corresponds to a point in a feature space. One can imagine that each dimension of that space corresponds to a semantic or grammatical characteristic of words. The hope is that functionally similar words get to be closer to each other in that space, at least along some directions. A sequence of words can thus be transformed into a sequence of these learned feature vectors. The neural network learns to map that sequence of feature vectors to the probability distribution over the next word in the sequence.

The distributed representation approach to LM has the advantage that it allows the model to generalize well to sequences that are not in the set of training word sequences, but that are similar in terms of their features, i.e., their distributed representation. Because neural networks tend to map nearby inputs to nearby outputs, the predictions corresponding to word sequences with similar features are mapped to similar predictions.

The above ideas of neural network LM have been implemented in various studies, some involving deep architecture. In 0 and Hinton, 2007), temporally factored RBM was used for language modeling. Unlike the traditional N-gram model the factored RBM uses distributed representations not only for context words but also for the words being predicted. This approach is generalized to deeper structures as reported in (Mnih and Hinton, 2008).

More recent work on neural network LM with deep architectures can be found in (Le et al., 2010, 2011; Mikolov et al., 2010). Use of hierarchical Bayesian priors in building up deep and recursive structure in LM appeared recently in (Huang and Renals, 2010)

D. Natural language processing and information retrieval

In the popular work on natural language processing, the authors of (Collobert and Weston, 2008) developed and employed a convolutional DBN as the common model to simultaneously solve a number of classic problems including part-of-speech tagging, chunking, named entity tagging, semantic role identification, and similar word identification. More recent work reported in (Collobert, 2010) further developed a fast purely discriminative approach for parsing based on the deep recurrent convolutional architecture called Graph Transformer Network.

A similar multi-task learning technique with DBN is used in (Deselaers et al., 2009) to attack a machine transliteration problem, which may be generalized to a more difficult machine translation problem.

The most interesting recent work on applying deep learning to natural language processing appears in (Socher et al., 2011), where a recursive neural network is used to build a deep architecture. The network is shown to be capable of successful merging of natural language words based on the learned semantic transformations of their original features. This deep learning approach provides an excellent performance on natural language parsing. The same approach is also demonstrated by the same authors to be successful in parsing natural scene images.

Finally, we discuss applications of DBN and deep auto encoder to document indexing and information retrieval (Salakhutdinov and Hinton, 2007). It is shown that the hidden variables in the last layer not only are easy to infer but also give a much better representation of each document, based on the word-count features, than the widely used latent semantic analysis. Using the compact code produced by deep networks, documents are mapped to memory addresses in such a way that semantically similar text documents are located at nearby address to facilitate rapid document retrieval. This idea is explored for audio document retrieval and some class of speech recognition problems with the initial exploration reported in (Deng et al, 2010).

VI. Summary and Discussions

This paper presents a brief history of deep learning, develops a categorization scheme to analyze the existing deep architectures in the literature into generative, discriminative, and hybrid classes. The DBN and DCN architectures are discusses in more detail, as they appear to be most popular and promising approaches. Applications of deep learning in four broad areas of information processing are then reviewed, including some of the author's own work with colleagues.

Deep learning is an emerging technology. Despite the empirical promising results reported so far, much needs to be developed. For example, recent published work shows that there is vast room to improve the current optimization techniques for learning deep architectures (Martens, 2010; Le et al., 2011; Martens and Sutskever, 2011). While the current learning strategy of generative pre-training followed by discriminative fine-tuning seems to work well empirically for many tasks, it fails to work for some other tasks that we have explored (e.g., language identification). For these tasks, the features extracted at the generative pre-training phase seem to describe the underlining speech variations well but do not contain sufficient information to distinguish between different languages. A learning strategy that can extract discriminative features is expected to provide better solutions. Extracting discriminative features may also greatly reduce the model size needed in the many current deep learning systems.

Further, effective and scalable parallel algorithms are essential to train deep models with very large data, as in many common information processing applications such as speech recognition and machine translation. The popular mini-batch stochastic gradient technique is difficult to be parallelized over computers. The common practice nowadays is to use graphical processing units (GPUs) to speed up the learning process. However, single machine GPU processing is not practical for large datasets, which is typical in speech recognition and similar applications. To make deep learning techniques scalable to thousands of hours of speech data, for example, theoretically sound parallel learning algorithms or novel architectures need to be developed. The DCN architecture presented in this paper is a promising direction toward the scalability goal, but much more work is needed in this area.

References

Baker, J., et. al. "Research developments and directions in speech recognition and understanding," *IEEE Sig. Proc. Mag.*, vol. 26, May 2009, pp. 75-80.

Bengio Y. "Learning deep architectures for AI," in Foundations and Trends in Machine Learning, Vol. 2, No. 1, 2009, pp. 1-127.

Bengio, Y., Ducharme, R., Vincent, P. and Jauvin, C. "A neural probabilistic language model," Proc. NIPS, 2000, pp. 933-938.

Bilmes, J. "Dynamic graphical models," IEEE Signal Processing Mag., vol. 33, pp. 29–42, 2010.

Bilmes, J. and Bartels, C. "Graphical model architectures for speech recognition," IEEE Signal Processing Mag., vol. 22, pp. 89–100, 2005.

Bridle, J., L. Deng, J. Picone, H. Richards, J. Ma, T. Kamm, M. Schuster, S. Pike, and R. Reagan, "An investigation fo segmental hidden dynamic models of speech coarticulation for automatic speech recognition," Final Report for 1998 Workshop on Langauge Engineering, CLSP, Johns Hopkins, 1998.

Collobert R. "Deep learning for efficient discriminative parsing," Proc. NIPS Workshop on Deep Learning and Unsupervised Feature Learning, 2010.

Collobert R. and Weston J. "A unified architecture for natural language processing: Deep neural networks with multitask learning," Proc. ICML, 2008.

Dahl, G., Yu, D., Deng, L., and Acero, A. "Context-dependent DBN-HMMs in large vocabulary continuous speech recognition," Proc. ICASSP, 2011.

Dahl, G., Yu, D., Deng, L., and Acero, A. "Context-dependent DBN-HMMs in large vocabulary continuous speech recognition," IEEE Trans. Audio, Speech, & Language Proc. Vol. 20 (1), January 2012.

Dahl, G., Ranzato, M., Mohamed, A. and Hinton, G. "Phone recognition with the mean-covariance restricted Boltzmann machine," Proc. NIPS, vol. 23, 2010, 469-477.

DARPA Deep Learning program, 2009--, http://www.darpa.mil/ipto/solicit/baa/BAA-09-40_PIP.pdf

Deng, L. DYNAMIC SPEECH MODELS --- Theory, Algorithm, and Application, Morgan & Claypool, December 2006.

Deng, L. "Computational Models for Speech Production," in Computational Models of Speech Pattern Processing, (NATO ASI Series), pp. 199-213, Springer Verlag, 1999.

Deng, L. and O'Shaughnessy, D. SPEECH PROCESSING --- A Dynamic and Optimization-Oriented Approach, Marcel Dekker, 2003.

Deng, L. and Yu, D. "Deep Convex Network: A scalable architecture for deep learning," Proc. Interspeech, 2011.

Deng, L., Yu, D. and Acero, A. "Structured speech modeling," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1492-1504, September 2006

Deng, L., Yu, D. and Acero, A. "A Bidirectional Target Filtering Model of Speech Coarticulation: two-stage Implementation for Phonetic Recognition," *IEEE Transactions on Audio and Speech Processing*, vol. 14, no. 1, pp. 256-265, January 2006.

Deng, L. and Yu, D. "Use of differential cepstra as acoustic features in hidden trajectory modeling for phonetic recognition, Proc. ICASSP, April 2007

Deng, L., Aksmanovic, M., Sun, D., and Wu, J. "Speech recognition using hidden Markov models with polynomial regression functions as nonstationary states," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 4, pp. 507-520, 1994.

Deng, L. "A generalized hidden Markov model with state-conditioned trend functions of time for the speech signal," Signal Processing, vol. 27, no. 1, pp. 65–78, 1992.

Deng, L. "A stochastic model of speech incorporating hierarchical nonstationarity," *IEEE Transactions on Speech and Audio Processing*, vol. 1, no. 4, pp. 471-475, 1993.

Deng L. and Sun, D. "A statistical approach to automatic speech recognition using the atomic speech units constructed from overlapping articulatory features," *Journal of the Acoustical Society of America*, vol. 85, no. 5, pp. 2702-2719, 1994.

Deng, L. and Sameti, H. "Transitional speech units and their representation by regressive Markov states: Applications to speech recognition," IEEE Transactions on speech and audio processing, vol. 4, no. 4, pp. 301–306, July 1996.

Deng, L., G. Ramsay, and D. Sun, "Production models as a structural basis for automatic speech recognition," Speech Communication, vol. 33, no. 2-3, pp. 93–111, Aug 1997.

Deng, L., "A dynamic, feature-based approach to the interface between phonology and phonetics for speech modeling and recognition," Speech Communication, vol. 24, no. 4, pp. 299–323, 1998.

Deng, L. "Switching dynamic system models for speech articulation and acoustics," in Mathematical Foundations of Speech and Language Processing, pp. 115–134. Springer-Verlag, New York, 2003.

Deng, L. "Expanding the scope of signal processing," *IEEE Signal Processing Magazine*, vol. 25, no. 3, May 2008.

Deng, L., Seltzer, M., Yu, D., Acero, A., Mohamed, A., and Hinton, G. "Binary coding of speech spectrograms using a deep auto-encoder," in Proc. *Interspeech, 2010*.

Deselaers T., Hasan S., Bender O. and Ney H. "A deep learning approach to machine transliteration," Proc. 4th Workshop on Statistical Machine Translation , pp. 233–241, Athens, Greece, March 2009.

Erhan, D., Bengio, Y., Courvelle, A., Manzagol, P., Vencent, P., and Bengio, S. "Why does unsupervised pre-training help deep learning?" J. Machine Learning Research, 2010, pp. 201-208.

Gibson, M. and Hain, T. "Error approximation and minimum phone error acoustic model estimation," IEEE Trans. Audio, Speech, and Language Proc., vol. 18, no. 6, August 2010, pp. 1269-1279.

Glorot, X., Bordes, A., and Bengio, Y. "Deep sparse rectifier neural networks," Proc. AISTAT, April 2011.

Glorot, X. and Bengio, Y. "Understanding the difficulty of training deep feedforward neural networks" Proc. AISTAT, 2010.

He, X., Deng, L., Chou, W. "Discriminative learning in sequential pattern recognition --- A unifying review for optimization-oriented speech recognition," *IEEE Sig. Proc. Mag.*, vol. 25, 2008, pp. 14-36.

He, X. and Deng, L. "Speech recognition, machine translation, and speech translation --- A unifying discriminative framework," IEEE Sig. Proc. Magazine, Vol. 28, November, 2011.

Hifny, Y. and Renals, S. "Speech recognition using augmented conditional random fields," IEEE Trans. Audio, Speech, and Language Proc., vol. 17, no. 2, February 2009, pp. 354-365.

Hinton, G., Krizhevsky, A., and Wang, S. "Transforming auto-encoders," Proc. Intern. Conf. Artificial Neural Networks, 2011.

Hinton, G. "A practical guide to training restricted Boltzmann machines," UTML Tech Report 2010-003, Univ. Toronto, August 2010.

Hinton, G., Osindero, S., and Teh, Y. "A fast learning algorithm for deep belief nets," Neural Computation, vol. 18, pp. 1527-1554, 2006.

Hinton, G. and Salakhutdinov, R. "Reducing the dimensionality of data with neural networks," Science, vol. 313. no. 5786, pp. 504 - 507, July 2006.

Huang, S. and Renals, S. "Hierarchical Bayesian language models for conversational speech recognition," IEEE Trans. Audio, Speech, and Language Proc., vol. 18, no. 8, November 2010, pp. 1941-1954.

Jaitly, N. and Hinton, G. "Learning a better representation of speech sound waves using restricted Boltzmann machines," *Proc. ICASSP*, 2011.

Jiang, H. and Li, X. "Parameter estimation of statistical models using convex optimization: An advanced method of discriminative training for speech and language processing," IEEE Signal Processing Magazine, vol. 27, no. 3, pp. 115–127, 2010.

Ketabdar, H. and Bourlard, H. "Enhanced phone posteriors for improving speech recognition systems," IEEE Trans. Audio, Speech, and Language Proc., vol. 18, no. 6, August 2010, pp. 1094-1106.

Le, H., Allauzen, A., Wisniewski, G., and Yvon, F. "Training continuous space language models: Some practical issues," in Proc. of EMNLP, 2010, pp. 778–788.

Le, H., Oparin, I., Allauzen, A., Gauvain, J., and Yvon, F. "Structured output layer neural network language model," Proc. ICASSP, 2011.

Le, Q., Ngiam, J., Coates, A., Lahiri, A., Prochnow, B., and Ng, A. "On optimization methods for deep learning," Proc. ICML, 2011.

Lecun Y., Chopra S., Ranzato, M., and Huang, F. "Energy-based models in document recognition and computer vision," Proc. Intern. Conf. Document Analysis and Recognition, (ICDAR), 2007.

Lee, C.-H. "From knowledge-ignorant to knowledge-rich modeling: A new speech research paradigm for next-generation automatic speech recognition," Proc. ICSLP, 2004, p. 109111.

Lee, H., Largman, Y., Pham, P., Ng, A. "Unsupervised feature learning for audio classification using convolutional deep belief networks," Proc. NIPS, Dec. 2009.

Martens J. "Deep learning with Hessian-free optimization," Proc. ICML, 2010.

Martens J. and Sutskever, I. "Learning recurrent neural networks with Hessian-free optimization," Proc. ICML, 2011.

Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., and Khudanpur, S. "Recurrent neural network based language model," Proc. ICASSP, 2011, 1045–1048.

Mnih A. and Hinton G. "Three new graphical models for statistical language modeling," Proc. ICML, 2007, pp. 641-648.

Mnih A. and Hinton G. "A scalable hierarchical distributed language model" Proc. NIPS, 2008, pp. 1081-1088.

Mohamed A., Yu, D., and Deng, L. "Investigation of full-sequence training of deep belief networks for speech recognition," Proc. Interspeech, Sept. 2010.

Mohamed, A., Dahl, G. and Hinton, G. "Acoustic Modeling Using Deep Belief Networks", IEEE Trans. Audio, Speech, & Language Proc. Vol. 20 (1), January 2012.

Mohamed, A., Dahl, G., and Hinton, G. "Deep belief networks for phone recognition," NIPS Workshop on deep learning for speech recognition, 2009.

Morgan, N. "Deep and Wide: Multiple Layers in Automatic Speech Recognition," IEEE Trans. Audio, Speech, & Language Proc. Vol. 20 (1), January 2012.

Morgan, N., Q. Zhu, A. Stolcke, K. Sonmez, S. Sivadas, T. Shinozaki, M. Ostendorf, P. Jain, H. Hermansky, D. Ellis, G. Doddington, B. Chen, O. Cretin, H. Bourlard, , and M. Athineos, "Pushing the envelope - aside [speech recognition]," IEEE Signal Processing Magazine, vol. 22, no. 5, pp. 81–88, Sep 2005.

Nair, V. and Hinton, G. "3-d object recognition with deep belief nets," Proc. NIPS, 2009.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. "Multimodal deep learning," Proc. ICML, 2011.

Ngiam, J., Chen, Z., Koh , P., and Ng, A. "Learning deep energy models," Proc. ICML, 2011.

Ostendorf, V. Digalakis, and O. Kimball, "From HMM's to segment models: A unified view of stochastic modeling for speech recognition," IEEE Trans. Speech and Audio Proc., vol. 4, no. 5, September 1996.

Picone, P., S. Pike, R. Regan, T. Kamm, J. bridle, L. Deng, Z. Ma, H. Richards, and M. Schuster, "Initial evaluation of hidden dynamic models on conversational speech," Proc. ICASSP, 1999.

Pinto, J., Garimella, S., Magimai-Doss, M., Hermansky, H., and Bourlard, H. "Analysis of MLP-based hierarchical phone posterior probability estimators," IEEE Trans. Audio, Speech, and Language Proc., vol. 19, no. 2, Feb. 2011.

Poon, H. and Domingos, P. "Sum-product networks: A new deep architecture," Proc. Twenty-Seventh Conference on Uncertainty in Artificial Intelligence, 2011. Barcelona, Spain.

Ranzato, M., Chopra, S. and LeCun, Y., and Huang, F.-J. "Energy-based models in document recognition and computer vision,'' *Proc. International Conference on Document Analysis and Recognition (ICDAR), 2007.*

Rennie, S., Hershey, H., and Olsen, P. "Single-channel multitalker speech recognition — Graphical modeling approaches," IEEE Signal Processing Mag., vol. 33, pp. 66–80, 2010.

Salakhutdinov R. and Hinton, G. "Semantic hashing," Proc. SIGIR Workshop on Information Retrieval and Applications of Graphical Models, 2007.

Salakhutdinov R. and Hinton, G. "Deep Boltzmann machines," Proc. AISTATS, 2009.

Saon G. and J.-T. Chien, J.-T. "Bayesian sensing hidden Markov models," IEEE Trans. Audio, Speech, & Language Proc. Vol. 20 (1), January 2012.

Sheikhzadeh, H. and Deng, L. "Waveform-based speech recognition using hidden filter models: Parameter selection and sensitivity to power normalization," IEEE Trans. on Speech and Audio Processing, Vol. 2, pp. 80-91, 1994.

Sivaram G. and Hermansky, H. "Sparse Multilayer Perceptron for Phoneme Recognition," IEEE Trans. Audio, Speech, & Language Proc. Vol. 20 (1), January 2012.

Socher, R., Lin, C., Ng, A., and Manning, C. "Learning continuous phrase representations and syntactic parsing with recursive neural networks," Proc. ICML, 2011.

Sutskever, I., Martens J., and Hinton, G. "Generating text with recurrent neural networks", Proc. ICML, 2011.

Taylor, G., Hinton, G. E., and Roweis, S. "Modeling human motion using binary latent variables." Proc. NIPS, 2007.

Tang, Y. and Eliasmith, C. "Deep networks for robust visual recognition," Proc. ICML, 2010.

Taralba, A, Fergus R, and Weiss, Y. "Small codes and large image databases for recognition," Proc. CVPR, 2008.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P. "Stacked denoising autoencoders: Leaning useful representations in a deep network with a local denoising criterion," J. Machine Learning Research, Vol. 11, 2010, pp. 3371-3408.

Wohlmayr, M., Stark, M., Pernkopf, F. "A probabilistic interaction model for multipitch tracking with factorial hidden Markov model," IEEE Trans. Audio, Speech, and Language Proc., vol. 19, no. 4, May. 2011.

Xiao, L. and Deng, L. A Geometric Perspective of Large-Margin Training of Gaussian Models, in *IEEE Signal Processing Magazine*, vol. 27, no. 6, pp. 118-123, IEEE, November 2010.

Yang, D., Furui, S., et al. "Combining a two-step CRF model and a joint source channel model for machine transliteration," Proc. ACL, Uppsala, Sweden, 2010, pp. 275-280.

Yu, D. and Deng, L. "Deep learning and its applications to signal and information processing," IEEE Signal Processing Magazine, January 2011, pp. 145-154.

Yu, D. and Deng, L. "Deep-structured hidden conditional random fields for phonetic recognition," Proc. Interspeech, Sept. 2010.

Yu, D, Deng, L., Gong, Y. and Acero, A. "A novel framework and training algorithm for variable-parameter hidden Markov models," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 17, no. 7, September 2009, pp. 1348-1360.

Yu D. and Deng, L. Solving nonlinear estimation problems using Splines , in *IEEE Signal Processing Magazine*, vol. 26, no. 4, pp. 86-90, July 2009.

Yu D. and Deng, L. "Accelerated parallelizable neural networks learning algorithms for speech recognition," Proc. *Interspeech 2011,*

Yu, D., Wang, S., Karam, Z., Deng, L. "Language recognition using deep-structured conditional random fields," Proc. ICASSP, April 2010, pp. 5030-5033.

Yu, D., Li, J.-Y., and Deng, L. "Calibration of confidence measures in speech recognition," IEEE Trans. Audio, Speech and Language, 2010.

Yu, D., Wang, S., Deng, L., "Sequential labeling using deep-structured conditional random fields", J. of Selected Topics in Signal Processing, 2010.

Zamora-Martínez, F., Castro-Bleda, M., España-Boquera, S. "Fast evaluation of connectionist language models," Intern. Conf. Artificial Neural Networks, 2009, pp. 144--151.

Zen, H., Nankaku, Y., and Tokuda, K. "Continuous stochastic feature mapping based on trajectory HMMs," IEEE Trans. Audio, Speech, and Language Proc., vol. 19, no. 2, Feb. 2011, pp. 417-430.