Deep Learning for Natural Language Processing

Tianchuan Du
Department of Computer and Information Sciences
University of Delaware
Newark, DE 19711
tdu@udel.edu

Vijay K. Shanker
Department of Computer and Information Sciences
University of Delaware
Newark, DE 19711
vijay@cis.udel.edu

Abstract

Deep learning has emerged as a new area of machine learning research. It tries to mimic the human brain, which is capable of processing and learning from the complex input data and solving different kinds of complicated tasks well. It has been successfully applied to several fields such as images, sounds, text and motion. The techniques developed from deep learning research have already been impacting the research of natural language process. This paper reviews the recent research on deep learning, its applications and recent development in natural language processing.

1 Introduction

Deep learning has emerged as a new area of machine learning research since 2006 (Hinton and Salakhutdinov 2006; Bengio 2009; Arel, Rose et al. 2010; Yoshua 2013). Deep learning (or sometimes called feature learning or representation learning) is a set of machine learning algorithms which attempt to learn multiple-layered models of inputs, commonly neural networks. The deep neural networks are composed of multiple levels of non-linear operations. Before 2006, searching the parameter space of deep architectures is a nontrivial task, but recently deep learning algorithms have been proposed to resolve this problem with notable success, beating the state-of-the-art in certain areas (Bengio 2009).

2 Deep learning

A central idea (Bengio, Courville et al. 2013) of deep learning is referred to as greedy layerwise unsupervised pre-training, which is to learn a hierarchy of features one level at a time. The features learning process can be purely unsupervised, which can take advantage of massive unlabeled data. The feature learning is trying to learn a new transformation of the previously learned features at each level, which is able to reconstruct the original data. The greedy layerwise unsupervised pre-training (Hinton, Osindero et al. 2006; Bengio, Lamblin et al. 2007; Bengio 2009) is based on training each layer with an unsupervised learning algorithm, taking the features produced at the previous level as input for the next level. It is then straightforward to extracted features either as input to a standard supervised machine learning predictor (such as an Support Vector Machines or Conditional Random Field) or as initialization for a deep supervised neural network. For example, each iteration of unsupervised feature learning adds one layer of weights to a deep neural network. Finally, the set of layers with learned weights could be stacked to initialize a deep supervised predictor, such as a neural network classifier, or a deep generative model, such as a Deep Boltzmann Machine (Salakhutdinov and Hinton 2009).

2.1 Stacked auto-encoder

One good illustration of the idea of greedy layerwise unsupervised pre-training is the stacked auto-encoder. An auto-encoder is an artificial neural network used for learning efficient coding (Liou, Huang et al. 2008). The aim of an auto-encoder is to learn a compressed representation (encoding) for a set of data, which means that it was being used for dimensionality reduction or data compression. As shown in Figure 1, the auto-encoder is consisted of an input layer, a number of considerably smaller hidden layers, which will form the encoding, and an output layer, which will try to reconstruct the input layer. It was shown that if linear neurons are used, or only a single sigmoid hidden layer, then the optimal solution to an auto-encoder is strongly related to PCA (Bourlard and Kamp 1988). Then use the learned feature to train another layer of auto-encoder. Finally, use the learned weights to initialize a deep neural network as shown in Figure 2.

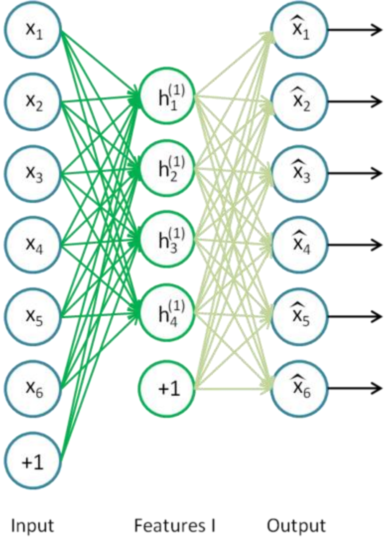

Figure 1. Structure of Auto-encoder. Set the output layer same as the input layer to train the network. The hidden layer is the learned feature of the input.

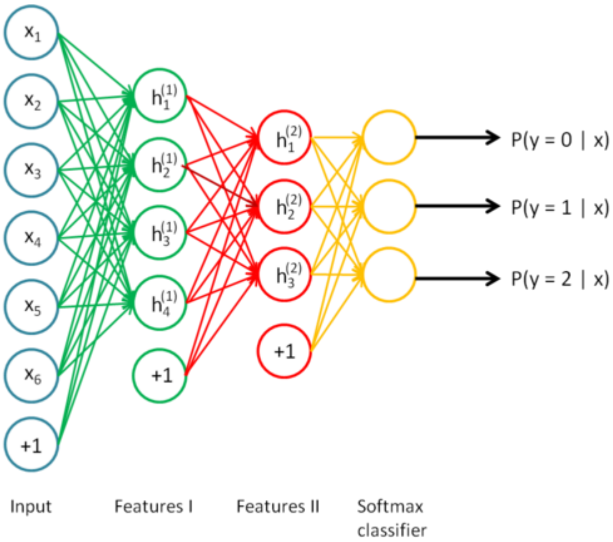

Figure 2. Stacked auto-encoder. Use the weights of auto-encoders to initialize the deep neural network. Then fine-tune the whole network by back propagation.

2.2 Deep Boltzmann Machines

Another way to implement the pre-training is through restricted Boltzmann machines (RBMs) as explained in Hinton's science paper (Hinton and Salakhutdinov 2006). It uses the learned restricted Boltzmann machines (RBMs) to try to regenerate the original input data. The learned feature activations of one RBM are used as the input data for training the next layer RBM in the stack. After the pre-training, the RBMs are "unrolled" to create a deep network, which is then fine-tuned using back-propagation of error derivatives as shown in Figure 3. The stacks of RBMs will create Deep Boltzmann Machines (Salakhutdinov and Hinton 2009). Then use the pre-trained DBM to initialize a deep neural network and train with back propagation as the stacked auto-encoder explained in the previous section.

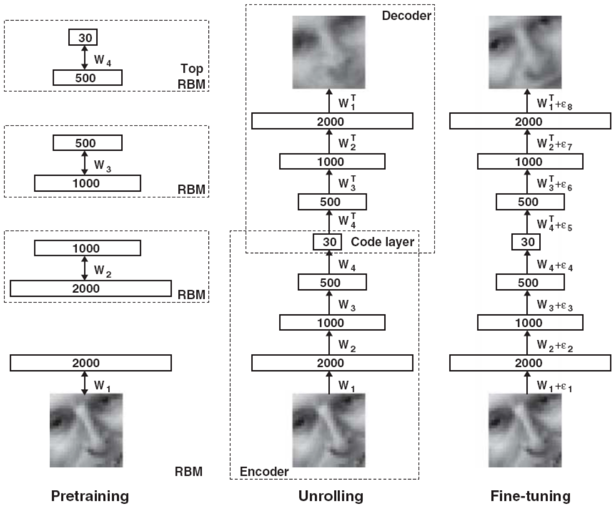

Figure 3. Restricted Boltzmann Machines to compress images.

2.3 Why deep?

One of the main reasons to go deep is that a non-linear function can be more efficiently represented by deep architecture with fewer parameters. The most formal arguments about the power of deep architectures come from investigations into computational complexity of circuits. The investigations suggests that when a function can be compactly represented by a deep architecture, it might need a very large architecture to be represented by an insufficiently deep one (Bengio 2009).

In another word, a number of computational complexity results strongly suggest that functions that can be compactly represented with a deeper architecture could require a very large number of elements in order to be represented by a shallower architecture. Because each parameter of the architecture might have to be selected or learned, using examples, these results suggest that depth of architecture can be very important from the point of view of statistical efficiency. Another reason is that deep representations might allow for a hierarchical representation. And multiple levels of latent variables allow combinatorial sharing of statistical strength (Bengio 2009).

Inspired by the architectural depth of the brain, neural network researchers had wanted for decades to train deep multi-layer neural networks (Utgoff and Stracuzzi 2002; Bengio and Lecun 2007), but it was not successful before 2006: researchers reported positive experimental results with typically two or three levels (i.e. one or two hidden layers), but training deeper networks consistently yielded poorer results. It was sometimes considered a breakthrough happened in 2006: Hinton and collaborators at University of Toronto introduced Deep Belief Networks or DBNs for short (Hinton, Osindero et al. 2006), with a learning algorithm that greedily trains one layer at a time, exploiting an unsupervised learning algorithm for each layer, a Restricted Boltzmann Machine (RBM)(Freund and Haussler 1994). Shortly after, related algorithms based on auto-encoders were proposed (Poultney, Chopra et al. 2006; Bengio, Lamblin et al. 2007), which apparently follows the same principle: guiding the training of intermediate levels of representation using unsupervised learning, which can be performed locally at each level. More other algorithms for deep architectures were proposed that exploit neither RBMs nor auto-encoders, but they followed the same principle (Mobahi, Collobert et al. 2009; Weston, Ratle et al. 2012).

2.4 Multi-Task and Transfer Learning, Domain Adaptation

Another advantage of deep learning is transfer learning. Transfer learning is the ability of a learning algorithm to exploit commonalities between different learning tasks in order to share statistical strength, and transfer knowledge across different tasks. As discussed below, it is hypothesized that feature learning algorithms have an advantage for such tasks because they learn features that capture underlying factors, a subset of features which may be relevant for a particular task, as illustrated in Figure 4. This hypothesis seems confirmed by a number of empirical results showing the advantages of feature learning or deep learning algorithms in domain adaptation and mult-task (Bengio, Courville et al. 2013).

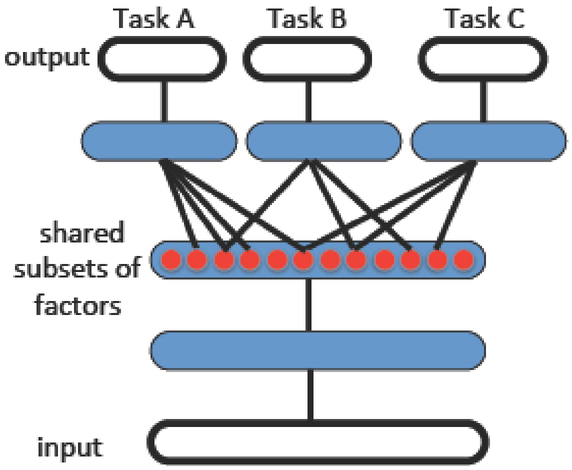

Fig. 4. Illustration of a feature learning model which discovers explanatory factors (the middle hidden layer in red). The shared features are learned unsupervised or supervised. Because these subsets overlap, sharing of statistical strength allows gains in generalization.

The illustrative empirical examples are the two transfer learning challenges held in 2011 and won by feature leaning or deep learning algorithms. The first one was the Transfer Learning Challenge, which held at an ICML 2011 workshop. It was won using unsupervised layer-wise pre-training (Bengio ; Mesnil, Dauphin et al. 2012). A second Transfer Learning Challenge was held at NIPS 2011's Challenges in Learning Hierarchical Models Workshop and also won by deep learning (Goodfellow, Courville et al. 2012). There more examples of the successful application of feature learning in fields related to transfer learning include domain adaptation (Glorot, Bordes et al. 2011; Chen, Xu et al. 2012).

3 The Applications of Deep Learning

During the past several years, the deep learning techniques have already been impacting a wide range of machine learning and artificial intelligence. It is thought that moving machine learning closer to one of its original goals: Artificial Intelligence. It has been successfully applied to several fields such as images, sounds, text and motion. The rapid increase in scientific activity on deep learning has been motivated by the empirical successes both in academia and in industry.

3.1 Object Recognition

Object recognition is thought to be a nontrivial task for computer. MNIST digit image classification problem has been used as benchmark for many machine learning algorithms, deep learning was focused on the problem since 2006 (Hinton, Osindero et al. 2006; Bengio, Lamblin et al. 2007), outperforming the supremacy of SVMs (1.4% error) on this dataset. The latest records are still held by deep networks: Ciresan et al. (Ciresan, Meier et al. 2012) currently claims the title of state-of-the-art for the unconstrained version of the task (e.g., using a convolutional architecture), with 0.27% error, and Rifai et al. (Rifai, Dauphin et al. 2011) is state-of-the-art for the knowledge free version of MNIST, with 0.81% error.

In the last few years, deep learning has extended from digits to object recognition in natural images. The latest breakthrough has been achieved on the ImageNet dataset, which improve the state-of-the-art error rate from 26.1% to 15.3% (Krizhevsky, Sutskever et al. 2012).

3.2 Speech Recognition and Signal Processing

Speech recognition was one of the early applications of neural networks, in particular convolutional (or time-delay) neural networks. The recent revival of interest in neural networks, deep learning, and representation learning has had a strong impact in the area of speech recognition. Deep learning was thought to yield breakthrough results (Dahl, Mohamed et al. 2010; Seide, Li et al. 2011; Dahl, Yu et al. 2012; Mohamed, Dahl et al. 2012), obtained by several academics as well as researchers at industrial labs, even bringing these algorithms to a larger scale and into products. For example, Microsoft has released a new version of their MAVIS (Microsoft Audio Video Indexing Service) speech system based on deep learning in 2012 (Seide, Li et al. 2011). In this paper, the author reduce the word error rate on four major benchmarks by about 30% (from 27.4% to 18.5% on RT03S) compared to state-of-the-art models based on Gaussian mixtures for the acoustic modeling and trained on the same amount of data (309 hours of speech). Similarly Dahl (Dahl, Yu et al. 2012) managed to decrease the relative error rate by between 16% and 23% on a smaller large-vocabulary speech recognition benchmark (Bing mobile business search dataset, with 40 hours of speech.

The standard deep neural network 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. One way to solve this problem is through the HMM.

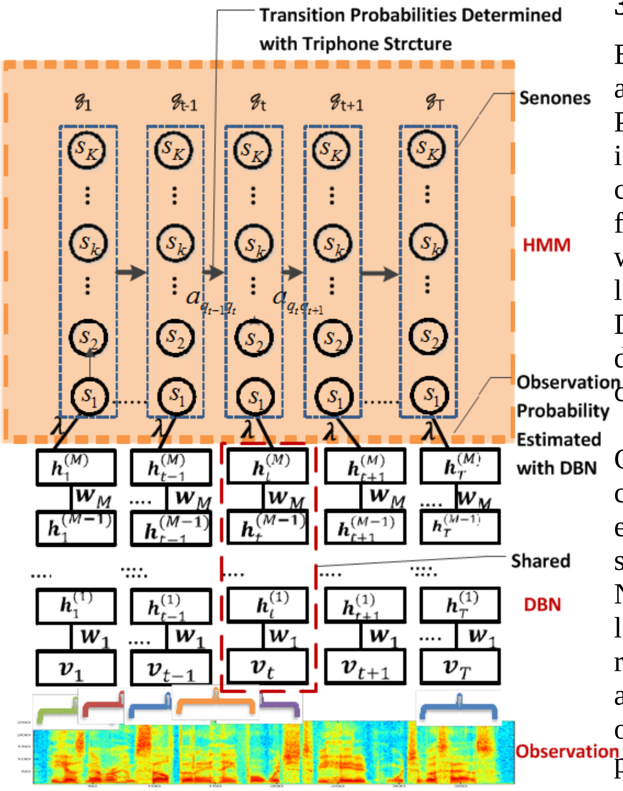
Transition Probabilities Determined with Triphone Strcture
Senones
HMM
Observation Probability Estimated with DBN
Shared
DBN
Observation

Figure 5: Interface between DBN/DNN and HMM to form a DBN-HMM or DNN-HMM. This architecture has been successfully used in speech recognition experiments reported in (Dahl et al., 2012).

The HMM is a convenient tool to model the sequence data with variable length, which based on dynamic programing operations. By integrating static classifiers and HMM, it is able to handle dynamic or sequential patterns. Thus, it is natural to combine deep neural network and HMM to bridge the gap between the static and sequence pattern recognition. A popular architecture to fulfill this is shown in Figure 5. This architecture has been successfully used in speech recognition experiments as reported in (Dahl et al., 2012).

Other approaches to tackle the problem that the dimensionality of both inputs and outputs are variable are based on recurrent neural networks or convolutional network (Collobert and Weston 2008; Socher, Huang et al. 2011; Socher, Pennington et al. 2011). They have also been applied to music, substantially beating the state-of-the-art in polyphonic transcription (Boulanger-Lewandowski, Bengio et al. 2012), with a relative error improvement of between 5% and 30% on a standard benchmark of four different datasets.

3.3 Natural Language Processing

Besides speech recognition, deep learning has been applied to many other Natural Language Processing applications. One important application is word embedding. The idea that symbolic data can be represented via distributed representation for was introduced by Hinton (Hinton 1986). It was first developed in the context of statistical language modeling by Bengio et al. (Bengio, Ducharme et al. 2003). The learning of a distributed representation for each word, also called a word embedding.

Collobert et al. (Collobert and Weston 2008; Collobert, Weston et al. 2011) applied deep convolutional network to implement the word embedding. He further developed the SENNA system that shares representations across different NLP tasks. This is also strong evidence that deep learning has the transfer learning potential. The result in this paper illustrated that the deep learning approaches surpasses the state-of-the-art on most of the tasks but is much faster than traditional predictors.

One major contribution of Collobert's work is to avoid task-specific, "man-made" feature engineering, and to learn versatility and unified features automatically from deep learning. Those learned features can be shared by all natural language processing tasks. The system described in (Collobert and Weston 2008; Collobert, Weston et al. 2011) automatically learns internal representation from vast amounts of mostly unlabeled training data (Deng and Yu ; Bengio, Courville et al. 2013). It defines a unified architecture for Natural Language Processing that learns features that are relevant to the many well-known NLP tasks including part-of-speech tagging, chunking, named-entity recognition, learning a language model and the task of semantic role-labeling given very limited prior knowledge. All of these tasks are integrated into a single system which is trained jointly. All the tasks except the language model are supervised tasks with labeled training data. The language model is trained in an unsupervised fashion on the entire Wikipedia website.

The theme behind the deep learning approach is different from the traditional NLP approach, which is: extract from the sentence a rich set of hand-designed features which are then fed to a classical shallow classification algorithm, e.g. a Support Vector Machine (SVM), often with a linear kernel. In this way, the choice of features is a completely empirical process, mainly based on trial and error, and the feature selection is task dependent, implying additional research for each new NLP task. It has some success for simple NLP tasks such as POS. But complex tasks like SRL then require a large number of possibly complex features (e.g., extracted from a parse tree) which makes such systems slow and intractable for large-scale applications. Instead in this paper, the author proposed a deep neural network (NN) architecture, trained in an end-to-end fashion. The input sentence is processed by several layers of feature extraction. The features in deep layers of the network are automatically trained by backpropagation to be relevant to the task. The structure is summarized in Figure 6. In this structure, the first layer extracts features for each word. The second layer extracts features from the sentence treating it as a sequence with local and global structure. The following layers are classical NN layers. The semi-supervised training of SRL using the language model performs better than other combinations. The result reported in this paper was as low as 14.30% in per-word error rate, which beats the state-of-the-art, 16.54%, based on parse trees (Pradhan et al., 2004). Besides, this system is the only one not to use POS tags or parse tree features. With the multiple task learning, the author managed to obtain 2.91% for POS and 3.8% for chunking. POS error rates in the 3% range are state-of-the-art.

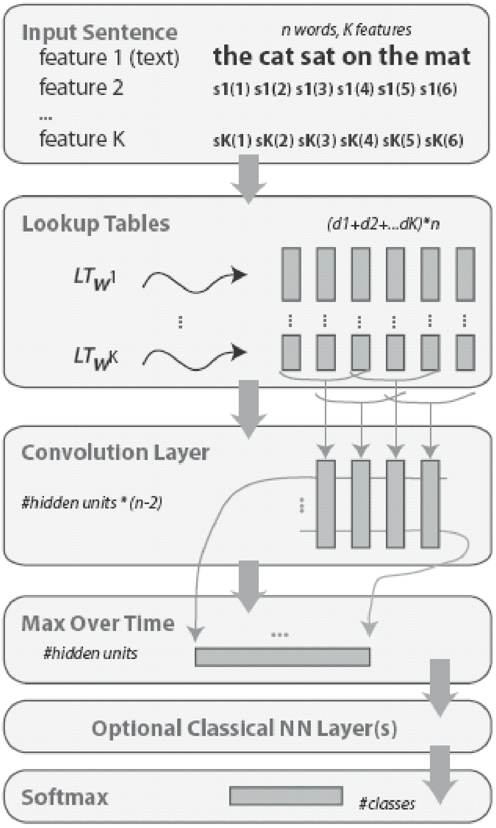

Figure 6. A general deep NN architecture for NLP reported in Collobert's work (Collobert and Weston 2008). Given an input sentence, the NN outputs class probabilities for one chosen word.

Beside the standard deep neural networks, Recurrence Neutral Network (RNN) is successfully applied to many aspects of natural language processing. Stanford NLP group recently applied RNN to sentiment analysis for semantic compositionality (Socher, Perelygin et al.). It improves the state of the art in single sentence positive/negative classification from 80% up to 85.4%. RNN was also applied to parsing, which improves the PCFG of the Stanford Parser by 3.8% to obtain an F1 score of 90.4% (Socher, Bauer et al. 2013). The neural network language model was also improved by adding recurrence to the hidden layers (Mikolov, Deoras et al. 2011), allowing it to surpass the state-of-the-art (smoothed n-gram models) not only in terms of perplexity (exponential of the average negative log-likelihood of predicting the right next word, going down from 140 to 102) but also in terms of word error rate in speech recognition, decreasing it from 17.2% (KN5 baseline) or 16.9% (discriminative language model) to 14.4% on the Wall Street Journal benchmark task. It have also been applied to statistical machine translation (Schwenk, Rousseau et al. 2012), which improves the BLEU score by almost 2 points. Similar structure, recursive auto-encoders (which generalize recurrent networks) have also been used to beat the state-of-the-art in full sentence paraphrase detection (Socher, Huang et al. 2011) almost doubling the F1 score for paraphrase detection. Deep learning can also be used to perform word sense disambiguation (Bordes, Glorot et al. 2012), improving the accuracy from 67.8% to 70.2%.

4 Conclusions

In sum, deep learning has becoming a new field of machine learning, and has gained extensive interests in different research area. It has shown some advantages over the traditional machine learning methods in some fields. Although deep learning works well in many machine learning tasks, it works equally poorly in some areas as the other learning methods. Besides most of the deep learning investigations are empirical, solid theoretical foundations of deep learning need to be established. Deep learning has been applied to natural language processing with some success. The result from deep learning looks promising, but the results are preliminary from some subfields of NLP, and from a few research groups. Besides, the result for NLP is still far from satisfying, letting the computers understand human languages. Further investigations are needed for both deep learning and NLP.

References

Arel, I., D. C. Rose, et al. (2010). "Deep Machine Learning - A New Frontier in Artificial Intelligence Research [Research Frontier]." Computational Intelligence Magazine, IEEE 5(4): 13-18.

Bengio, Y. "Deep Learning of Representations for Unsupervised and Transfer Learning."

Bengio, Y. (2009). "Learning Deep Architectures for AI." Found. Trends Mach. Learn. 2(1): 1-127.

Bengio, Y., A. Courville, et al. (2013). "Representation learning: A review and new perspectives."

Bengio, Y., R. Ducharme, et al. (2003). "A Neural Probabilistic Language Model." Journal of Machine Learning Research 3: 1137-1155.

Bengio, Y., P. Lamblin, et al. (2007). "Greedy layer-wise training of deep networks." Advances in neural information processing systems 19: 153.

Bengio, Y. and Y. Lecun (2007). Scaling learning algorithms towards AI. Large-Scale Kernel Machines. L. Bottou, O. Chapelle, D. Decoste and J. Weston, MIT Press.

Bordes, A., X. Glorot, et al. (2012). Joint learning of words and meaning representations for open-text semantic parsing. International Conference on Artificial Intelligence and Statistics.

Boulanger-Lewandowski, N., Y. Bengio, et al. (2012). "Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription." arXiv preprint arXiv:1206.6392.

Bourlard, H. and Y. Kamp (1988). "Auto-association by multilayer perceptrons and singular value decomposition." Biological cybernetics 59(4-5): 291-294.

Chen, M., Z. Xu, et al. (2012). "Marginalized denoising autoencoders for domain adaptation." arXiv preprint arXiv:1206.4683.

Ciresan, D., U. Meier, et al. (2012). Multi-column deep neural networks for image classification. Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE.

Collobert, R. and J. Weston (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. Proceedings of the 25th international conference on Machine learning, ACM.

Collobert, R., J. Weston, et al. (2011). "Natural language processing (almost) from scratch." The Journal of Machine Learning Research 12: 2493-2537.

Dahl, G., A.-r. Mohamed, et al. (2010). Phone recognition with the mean-covariance restricted Boltzmann machine. Advances in neural information processing systems.

Dahl, G. E., D. Yu, et al. (2012). "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition." Audio, Speech, and Language Processing, IEEE Transactions on 20(1): 30-42.

Deng, L. and D. Yu "DEEP LEARNING FOR SIGNAL AND INFORMATION PROCESSING."

Freund, Y. and D. Haussler (1994). Unsupervised learning of distributions of binary vectors using two layer networks, Computer Research Laboratory [University of California, Santa Cruz.

Glorot, X., A. Bordes, et al. (2011). Domain adaptation for large-scale sentiment classification: A deep learning approach. Proceedings of the 28th International Conference on Machine Learning (ICML-11).

Goodfellow, I. J., A. Courville, et al. (2012). "Spike-and-slab sparse coding for unsupervised feature discovery." arXiv preprint arXiv:1201.3382.

Hinton, G. E. (1986). Learning distributed representations of concepts. Proceedings of the eighth annual conference of the cognitive science society, Amherst, MA.

Hinton, G. E., S. Osindero, et al. (2006). "A Fast Learning Algorithm for Deep Belief Nets." Neural Computation 18(7): 1527-1554.

Hinton, G. E. and R. R. Salakhutdinov (2006). "Reducing the Dimensionality of Data with Neural Networks." Science 313(5786): 504-507.

Krizhevsky, A., I. Sutskever, et al. (2012). Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems 25.

Liou, C.-Y., J.-C. Huang, et al. (2008). "Modeling word perception using the Elman network." Neurocomputing 71(16): 3150-3157.

Mesnil, G., Y. Dauphin, et al. (2012). "Unsupervised and Transfer Learning Challenge: a Deep Learning Approach." Journal of Machine Learning Research-Proceedings Track 27: 97-110.

Mikolov, T., A. Deoras, et al. (2011). Empirical Evaluation and Combination of Advanced Language Modeling Techniques. INTERSPEECH.

Mobahi, H., R. Collobert, et al. (2009). Deep learning from temporal coherence in video. Proceedings of the 26th Annual International Conference on Machine Learning, ACM.

Mohamed, A.-r., G. E. Dahl, et al. (2012). "Acoustic modeling using deep belief networks." Audio, Speech, and Language Processing, IEEE Transactions on 20(1): 14-22.

Poultney, C., S. Chopra, et al. (2006). Efficient learning of sparse representations with an energy-based model. Advances in neural information processing systems.

Rifai, S., Y. N. Dauphin, et al. (2011). The manifold tangent classifier. Advances in neural information processing systems.

Salakhutdinov, R. and G. E. Hinton (2009). Deep boltzmann machines. International Conference on Artificial Intelligence and Statistics.

Schwenk, H., A. Rousseau, et al. (2012). Large, pruned or continuous space language models on a GPU for statistical machine translation. Proceedings of the NAACL-HLT 2012 Workshop: Will We Ever Really Replace the N-gram Model? On the Future of Language Modeling for HLT, Association for Computational Linguistics.

Seide, F., G. Li, et al. (2011). Conversational Speech Transcription Using Context-Dependent Deep Neural Networks. INTERSPEECH.

Socher, R., J. Bauer, et al. (2013). Parsing with compositional vector grammars. Proceedings of the ACL conference (to appear).

Socher, R., E. H. Huang, et al. (2011). Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. Advances in neural information processing systems.

Socher, R., J. Pennington, et al. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics.

Socher, R., A. Perelygin, et al. "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank."

Utgoff, P. E. and D. J. Stracuzzi (2002). "Many-layered learning." Neural Comput. 14(10): 2497-2529.

Weston, J., F. Ratle, et al. (2012). Deep learning via semi-supervised embedding. Neural Networks: Tricks of the Trade, Springer: 639-655.

Yoshua, B. (2013). "Representation Learning: A Review and New Perspectives." IEEE Transactions on Pattern Analysis and Machine Intelligence 35(8): 1798-1828.