Speech Recognition Using Deep Learning Algorithms

Yan Zhang, SUNet ID: yzhang5
Instructor: Andrew Ng

Abstract: Automatic speech recognition, translating of spoken words into text, is still a challenging task due to the high viability in speech signals. Deep learning, sometimes referred as representation learning or unsupervised feature learning, is a new area of machine learning. Deep learning is becoming a mainstream technology for speech recognition and has successfully replaced Gaussian mixtures for speech recognition and feature coding at an increasingly larger scale. The main target of this course project is to applying typical deep learning algorithms, including deep neural networks (DNN) and deep belief networks (DBN), for automatic continuous speech recognition.

1. Introduction

Automatic speech recognition, translating of spoken words into text, is still a challenging task due to the high viability in speech signals. For example, speakers may have different accents, dialects, or pronunciations, and speak in different styles, at different rates, and in different emotional states. The presence of environmental noise, reverberation, different microphones and recording devices results in additional variability.

Conventional speech recognition systems utilize Gaussian mixture model (GMM) based hidden Markov models (HMMs) [1, 2] to represent the sequential structure of speech signals. HMMs are used in speech recognition because a speech signal can be viewed as a piecewise stationary signal or a short-time stationary signal. In a short time-scale, speech can be approximated as a stationary process. Speech can be thought of as a Markov model for many stochastic purposes.Typically, each HMM state utilizes a mixture of Gaussian to model a spectral representation of the sound wave. HMMs-based speech recognition systems can be trained automatically and are simple and computationally feasible to use. However, one of the main drawbacks of Gaussian mixture models is that they are statistically inefficient for modeling data that lie on or near a non-linear manifold in the data space.

Neural networks trained by back-propagation error derivatives emerged as an attractive acoustic modeling approach for speech recognition in the late 1980s. In contrast to HMMs, neural networks make no assumptions about feature statistical properties. When used to estimate the probabilities of a speech feature segment, neural networks allow discriminative training in a natural and efficient manner. However, in spite of their effectiveness in classifying short-time units such as individual phones and isolated words, neural networks are rarely successful for continuous recognition tasks, largely because of their lack of ability to model temporal dependencies. Thus, one alternative approach is to use neural networks as a pre-processing e.g. feature transformation, dimensionality reduction for the HMM based recognition.

Deep learning [6-9], sometimes referred as representation learning or unsupervised feature learning, is a new area of machine learning. Deep learning is becoming a mainstream technology for speech recognition [10-17] and has successfully replaced Gaussian mixtures for speech recognition and feature coding at an increasingly larger scale. In the course project, we focus on deep belief networks (DBNs) for speech recognition. The main goal of this course project can be summarized as:

1) Familiar with end-to-end speech recognition process.
2) Review state-of-the-art speech recognition techniques.
3) Learn and understand deep learning algorithms, including deep neural networks (DNN), deep belief networks (DBN), and deep auto-encoders (DAE).
4) Applying deep learning algorithms to speech recognition and compare the speech recognition performance with conventional GMM-HMM based speech recognition method.

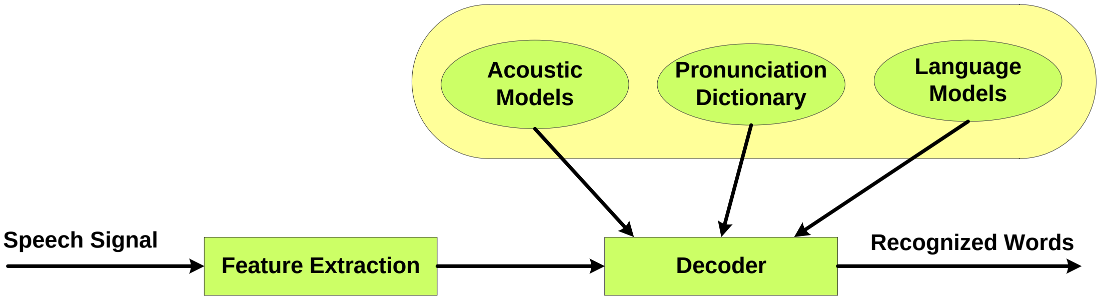

Fig. 1 A typical system architecture for automatic speech recognition

2. Automatic Speech Recognition System Model

The principal components of a large vocabulary continuous speech recognizer [1][2] are illustrated in Fig. 1. The input audio waveform from a microphone is converted into a sequence of fixed size acoustic vectors $\boldsymbol{Y} = [y_1, \cdots, y_T]$. This process is called feature extraction. The decoder then attempts to find the sequence of words $\boldsymbol{W} = [w_1, \cdots, w_L]$ which is most likely to have generated Y, i.e. the decoder tries to find

$$\widehat{W} = \underbrace{argmax}_{w} \{P(\boldsymbol{W}|\boldsymbol{Y})\}$$

However, since $P(\boldsymbol{W}|\boldsymbol{Y})$ is difficult to model directly, Bayes' Rule is used to transform the above equation into the equivalent problem of finding:

$$\widehat{W} = \underbrace{argmax}_{w} \{P(\boldsymbol{Y}|\boldsymbol{W})P(\boldsymbol{W})\}$$

The likelihood $P(\boldsymbol{Y}|\boldsymbol{W})$ is determined by an acoustic model and the prior $P(\boldsymbol{W})$ is determined by a language model.

For any given $\boldsymbol{W}$, the corresponding acoustic model is synthesized by concatenating phone models to make words as defined by a pronunciation dictionary. The parameters of these phone models are estimated from training data consisting of speech waveforms and their orthographic transcriptions. The language model is typically an N-gram model in which the probability of each word is conditioned only on its $N - 1$ predecessors. The *N*-gram parameters are estimated by counting *N*-tuples in appropriate text corpora. The decoder operates by searching through all possible word sequences using pruning to remove unlikely hypotheses thereby keeping the search tractable. When the end of the utterance is reached, the most likely word sequence is output. Alternatively, modern decoders can generate lattices containing a compact representation of the most likely hypotheses.

a. Feature Extraction

In automatic speech recognition, it is common to extract a set of features from speech signal. Classification is carried out on the set of features instead of the speech signals themselves. The feature extraction stage seeks to provide a compact representation of the speech waveform. This form should minimise the loss of information that discriminates between words, and provide a good match with the distributional assumptions made by the acoustic models. A popular feature vector Mel-frequency cepstral coefficients (MFCC), which provides a compact speech signal representation that are the results of a cosine transform of the real logarithm of the short-term energy spectrum expressed on a mel-frequency scale. MFCC coefficients are generated by applying a truncated discrete cosine transformation (DCT) to a log spectral estimate computed by smoothing an FFT with around 20 frequency bins distributed non-linearly across the speech spectrum. The nonlinear frequency scale used is called a mel scale and it approximates the response of the human ear. The DCT is applied in order to smooth the spectral estimate and approximately decorrelate the feature elements. After the cosine transform the first element represents the average of the log-energy of the frequency bins. This is sometimes replaced by the log-energy of the frame, or removed completely.

b. Hidden Markov Models

Predominantly, HMMs are used in ASR. A HMM is a stochastic finite state automaton built from a finite set of possible states $Q = \{q_1, \cdots, q_K\}$ with instantaneous transitions with certain probabilities between these states. Each of these states is associated with a specific emission probability distribution $p(x_n|q_k)$. Thus, HMMs can be used to model a sequence X of feature vectors as a piecewise stationary process where each stationary segment is associated with a specific HMM state. This approach defines two concurrent stochastic processes: the sequence of HMM-states modeling the temporal dynamics of speech, and a set of state output processes modeling the locally stationary property of the speech signal.

In speech recognition, we have to find the HMM M^* which maximizes the posterior probability $p(M|X)$ of the hypothesized HMM M given a sequence X of feature-vectors. Since this probability cannot be computed directly, it is usually split using Bayes' rule into the acoustic model (likelihood) $p(X|M)$ and a prior $p(M)$ representing the language model: $p(M|X) \propto p(X|M)p(M)$.

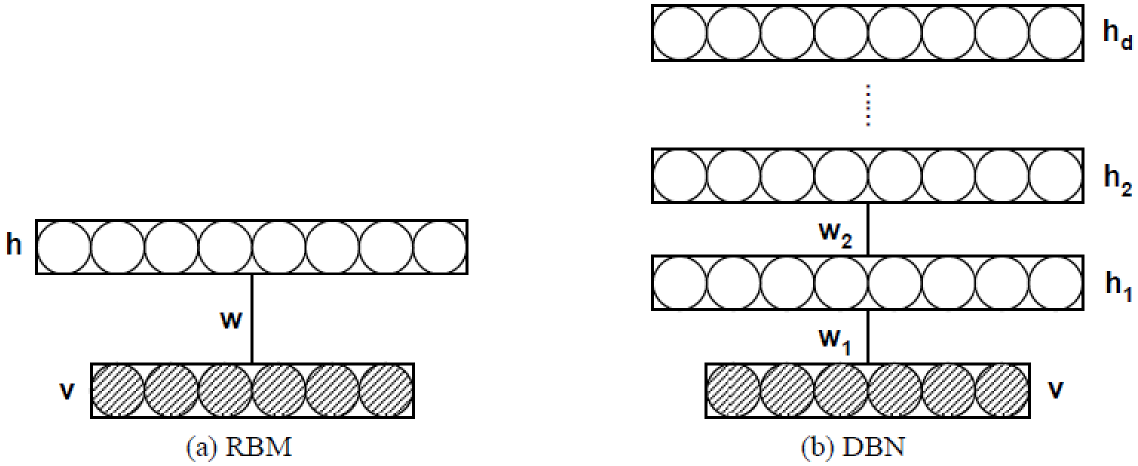

Fig.2: The DBN is composed of RBMs.

3. Deep Belief Networks

Deep Belief Networks (DBNs) are neural networks consisting of a stack of restricted Boltzmann machine (RBM) layers that are trained one at a time, in an unsupervised fashion to induce increasingly abstract representations of the inputs in subsequent layers.

3.1 Restricted Boltzmann Machines (RBMs) and Training

As shown in Fig. 2 (a), Each RBM has an input layer (visible layer) and a hidden layer of stochastic binary units. Visible and hidden layers are connected with a weight matrix and no connections exist between units in the same layer. Signal propagation can occur in two ways: recognition, where visible activations propagate to the hidden units; and reconstruction, where hidden activations propagate to visible units. The same weight matrix (transposed) is used for both recognition and reconstruction. By minimizing the difference between the original input and its reconstruction (i.e. reconstruction error) through a procedure called contrastive divergence (CD), the weights can be trained to generate the input patterns presented to the RBM with high probability. The RBM pretraining procedure of a DBN can be used to initialize the weights of a deep neural network, which can then be discriminatively fine-tuned by back-propagating error derivatives. The "recognition" weights of the DBN become the weights of a standard neural network. In cases where the RBM models the joint distribution of visible data and class labels, a hybrid training procedure can be used to fine-tune the generatively trained parameters.

3.2 DBN structure

Fig. 2 (b) shows the structure of a DBN. A DBN consists of a stack of RBMs, trained one at a time. Each layer of hidden units learns to represent features that capture higher order correlations in the original input data. In DBNs, subsequent layers usually decrease in size in order to force the network to learn increasingly compact representations of its inputs. The training procedure is sometimes augmented to optimize additional terms, such as the L1 and L2 norms of the weight matrices, or sparsity constraints on the unit activations. Weights are initialized from a normal distribution with zero mean and small standard deviation. Weight updates are applied after the presentation of a number of samples in a minibatch. After a number of training cycles through the full training dataset, the stack of RBMs is unfolded, such that first recognitions are computed through all subsequent layers, and next reconstructions through all layers in reverse order. The recognition and reconstruction weights are uncoupled, and can then be fine-tuned with gradient descent, either to become better at reconstructing the inputs, or — in combination with other supervised or reinforcement learning methods — to form features relevant to the task at hand.

3.3 Applying DBNs for Speech Recognition

To apply DBNs with fixed input and output dimensionality to phone recognition, a context window of n successive frames of feature vectors is used to set the states of the visible units of the lower layer of the DBN which produces a probability distribution over the possible labels of the central frame. To generate speech sequences, a sequence of probability distributions over the possible labels for each frame are fed into a standard Viterbi decoder.

4. Performance Evaluation

TIMIT acoustic-phonetic continuous speech corpus dataset [18] is used for performance evaluation. The speech was analyzed using a 25-ms Hamming window with 10-ms between the left edges of successive frames. The data were normalized to have zero mean and unit variance over the entire corpus. All experiments used a context window of 11 frames as the visible states. The 12th-order Mel frequency cepstral coefficients (MFCCs) and energy, along with their first and second temporal derivatives are extracted as speech features. The word error rate for GMM-HMM based speech recognition system is about 35%. The word error rate for a Deep Neural Network Hidden Markov Models (DNN-HMMs) speech recognition system with five hidden layers is around 32%. The word error rate of a DBN speech recognition system with three hidden layers and 2048 hidden units per layer is about 24%.

5. Conclusion

In this course project, typical deep learning algorithms, including deep neural networks (DNN), and deep belief networks (DBN) have been learned understood. Further, a DBN has been implemented for automatic speech recognition. The speech recognition performance evaluations on three speech recognition systems, namely, GMM-HMM, DNN-HMM and DBN, have been performed with TIMIT acoustic-phonetic continuous speech corpus dataset in terms of word error rate. The results have shown that the DBN-based speech recognition system beats other two speech recognition systems.

References:

[1]. S. Young, “Large Vocabulary Continuous Speech Recognition: A Review,” *IEEE Signal Processing Magazine*, vol. 13, no. 5, pp. 45–57, 1996.

[2]. J. Baker, L. Deng, J. Glass, S. Khudanpur, Chin hui Lee, N. Morgan, and D. O’Shaughnessy, “Developments and directions in speech recognition and understanding, part 1,” *Signal Processing Magazine, IEEE*, vol. 26, no. 3, pp. 75–80, may 2009.

[3]. Hinton, G., Osindero, S., and Teh, Y. “A fast learning algorithm for deep belief nets,” *Neural Computation*, vol. 18, pp. 1527-1554, 2006.

[4]. Yoshua Bengio, Pascal Lamblin, Dan Popovici and Hugo Larochelle,Greedy Layer-Wise Training of Deep Networks, in J. Platt et al. (Eds), *Advances in Neural Information Processing Systems 19 (NIPS 2006),* pp. 153-160, MIT Press, 2007.

[5]. Marc'Aurelio Ranzato, Christopher Poultney, Sumit Chopra and Yann LeCunEfficient Learning of Sparse Representations with an Energy-Based Model, in *J. Platt et al. (Eds), Advances in Neural Information Processing Systems (NIPS 2006)*, MIT Press, 2007.

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

[7]. Bengio Y, "Deep learning of representations: looking forward," in: *Statistical Language and Speech Processing*, pp. 1--37, Springer, 2013.

[8]. Bengio Y., Courville, A., and Vincent, P. "Representation learning: A review and new perspectives," *IEEE Trans. PAMI*, 2013a.

[9]. Li Deng, "A Tutorial Survey of Architectures, Algorithms, and Applications for Deep Learning" to appear in *APSIPA Transactions on Signal and Information Processing*, Cambridge University Press, 2014.

[10]. Mohamed, A., Dahl, G., and Hinton, G. "Deep belief networks for phone recognition," in *Proc. NIPS Workshop Deep Learning for Speech Recognition and Related Applications*, 2009.

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

[12]. G. Dahl, D. Yu, L. Deng, and A. Acero, "Large vocabulary continuous speech recognition with context-dependent DBN-HMMs," *ICASSP*, 2011.

[13]. G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large vocabulary speech recognition," *IEEE Trans. Audio, Speech, Lang. Proc.*, vol. 20, pp. 30–42, 2012.

[14]. Mohamed, A., Dahl, G. and Hinton, G. "Acoustic modeling using deep belief networks", *IEEE Trans. Audio, Speech, & Language Proc*. Vol. 20 (1), January 2012.

[15]. Mohamed, A., Hinton, G., and Penn, G., "Understanding how deep belief networks perform acoustic modelling," *Proc. ICASSP*, 2012.

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

[17]. Li Deng, Jinyu Li, Jui-Ting Huang, Kaisheng Yao, Dong Yu, Frank Seide, Michael Seltzer, Geoff Zweig, Xiaodong He, Jason Williams, Yifan Gong, and Alex Acero, "Recent Advances in Deep Learning for Speech Research at Microsoft", in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2013.

[18]. J.S. Garofolo, L.F. Lamel, W.M. Fisher, J.G. Fiscus, D.S. Pallett, N.L. Dahlgren, DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus, U.S. Dept. of Commerce, NIST, Gaithersburg, USA, 1993.