Never-Ending Learning with Dynamic Hidden Markov Network

Konstantin Markov[1,2], *Satoshi Nakamura*[1,2]

[1]Spoken Language Communication Group, NICT, Japan
[2]Spoken Language Communication Research Labs, ATR, Japan
konstantin.markov@nict.go.jp, satoshi.nakamura@nict.go.jp

Abstract

Current automatic speech recognition systems have two distinctive modes of operation: training and recognition. After the training, system parameters are fixed, and if a mismatch between training and testing conditions occurs, an adaptation procedure is commonly applied. However, the adaptation methods change the system parameters in such a way that previously learned knowledge is irrecoverably destroyed. In searching for a solution to this problem and motivated by the results of recent neuro-biological studies, we have developed a network of hidden Markov states that is capable of unsupervised on-line adaptive learning while preserving the previously acquired knowledge. Speech patterns are represented by state sequences or paths through the network. The network can detect previously unseen patterns, and if such a new pattern is encountered, it is learned by adding new states and transitions to the network. Paths and states corresponding to spurious events or "noises" and, therefore, rarely visited, are gradually removed. Thus, the network can grow and shrink when needed, i.e. it dynamically changes its structure. The learning process continues as long as the network lasts, i.e. theoretically forever, so it is called *never-ending learning*. The output of the network is the best state sequence and the decoding is done concurrently with the learning. Thus the network always operates in a single learning/decoding mode. Initial experiments with a small database of isolated spelled letters showed that the Dynamic Hidden Markov network is indeed capable of never-ending learning and can perfectly recognize previously learned speech patterns.

Index Terms: never-ending learning, life-long learning, dynamic hidden markov network, self-organization, topology representation.

1. Introduction

From a biological, as well as a technical, viewpoint, the artificial separation of a lifespan into a learning and recognition phase is a shortcoming of current automatic speech recognition (ASR) systems. While this approach is possible for systems that operate in a matched environment, it fails if the environment changes. To avoid costly retraining, recent research has focused on fast adaptation and on-line adaptive learning. However, such methods inevitably destroy previously well-learned patterns, a phenomenon known as *catastrophic forgetting* in cognitive science [1]. Besides only adapting to a changing environment, an intelligent system should also be able to preserve its knowledge. This suggests a life-long or never-ending learning capability without catastrophic forgetting [2]. Of course, gradual interference (knowledge erasure) is unavoidable and even desirable, since otherwise, soon or later such a system would exhaust its memory resources. In real applications, we rarely have control over the environments or prior knowledge about their characteristics. This leads to another requirement for the system: It should be able to perform unsupervised adaptive learning, which is called *self-organization* in neural network literature [3].

The main goal of current ASR systems is to find the most probable word sequence given the speech signal. In other words, we are interested only in the lexical information conveyed by the signal and any other existing information such as speaker identity (ID), speaking style, emotional state, etc. is considered as "noise" that causes unwanted variations in the signal characteristics. This requires a system that is robust against such variations. A lot of research has been done on improving the robustness of ASR systems and numerous methods and algorithms have been proposed. Still, there is no efficient solution to this problem that works consistently in all possible situations. When it comes to building machines capable of natural communication with humans, not only the speech lexical content, but also the speaker (ID, accent, emotions) and environment (office, street, etc.) information becomes important. Currently, to get such information, we build separate systems that can usually recognize or identify only a single factor, for example, only the speaker ID or the spoken language. In this case, the variability coming from the linguistic content is "unwanted" and has to be dealt with. The alternative is to design a system that instead of trying to normalize or reduce the speech signal variability is capable of learning it and outputting simultaneously not only the lexical but any other information we are interested in. Such a system should be able to learn continuously in an unsupervised manner, since it is impossible to have prior knowledge about the all possible variability sources. This again leads to the idea of having a self-organizing never-ending learning system.

In trying to bridge the gap between the learning abilities of human and machines, many researchers have turned to studies of human capabilities as a source of ideas for designing such systems [4]. Based on our everyday experience, we can say that humans are capable of learning throughout their life and that the acquisition of new knowledge does not wash away memories of prior learning. While much of how the human brain works is still not well understood, some basic principles of learning at the neuronal level, such as the Hebbian rule [5], have been formulated. Brain studies have shown that the nervous system has a topological structure - similar stimuli activate topologically close areas in the brain, and this observation has inspired the development of several neural network architectures [6, 7].

The never-ending or life-long learning principle gives rise to the so-called *stability-plasticity* dilemma [8] - How can a system preserve its previously learned knowledge while continuing to learn new things? Several solutions to this problem have been proposed in the neural networks research field, including Adaptive Resonance Theory (ART) [9], Life-long Learning Cell Structures [2] and Self-Organizing Incremental Neu-

ral Network [10]. Commonly, network plasticity is ensured by adding new nodes to accommodate the new knowledge, while decreasing learning rates for the connection weights provides the necessary network stability. Unfortunately, such neural networks do not work with spatio-temporal data such as speech patterns. A system that can simultaneously learn and recognize spatio-temporal patterns as well as recall them was proposed in [11]. The system is a combination of a self-organizing map (SOM) and an ART network that only takes input patterns of similar finite length. In addition, an off-line pre-processing step is required to learn the first SOM layer, which determines the system operating range in the input space. A never-ending learning system based on the so-called Guided Propagation Networks (GPNs) was demonstrated in [12]. Various possible applications of this system, including speech and natural language processing, were presented, but major shortcoming of the GPN is the need to transform the spatio-temporal input data into binary patterns.

In designing the Dynamic Hidden Markov network (DHMnet), which is the subject in this paper, we tried to implement the never-ending learning principle and avoid the limitations of the existing life-long learning structures. In doing so, our goal was to create a self-organizing, topology representing, never-ending learning speech model. We have to note that the DHMnet is far from being a full recognition system. It is intended to be the basic building block of a new type speech recognition system that is entirely based on the never-ending learning principle. In such system, the DHMnet would play a role similar to the HMM's role in the current generation of ASR systems.

2. The Dynamic Hidden Markov Network

2.1. General structure

The DHMnet consists of hidden Markov states with self-loops and transitions between them. Additionally, neighboring states are connected with lateral connections (more details are given in Section 2.4). Each state represents a part of the input feature space modeled by a multivariate Gaussian function. State sequences or paths through the network correspond to learned speech patterns or classes of patterns. Similarly to other approaches, network plasticity is ensured by adding new states and transitions whenever a new pattern is encountered. The practical problem is to define what should be considered as a "new" pattern and how to detect it. Inevitably, spurious events and noises would allocate states that may never be visited again. Such states (and paths) are considered "dead" and will be gradually removed from the network. The schematic structure of the DHMnet is shown in Fig.1, where transitions of a learned path are represented by directed solid lines, new paths with directed short dashed lines, and "dead" paths with directed long dashed lines. Undirected dashed lines represent lateral connections between states.

2.2. "Novelty" detection

Generally, any pattern that is sufficiently different from those that have been already learned can be considered a new pattern. To decide what is sufficiently different, we again turn to studies on the human auditory system. It is known that a human's sensitivity to changes in sound pressure level is limited. It has been found that for wide-band noise the smallest detectable change in intensity ΔI is approximately proportional to the intensity of the stimulus I. That is, $\Delta I/I$, is constant (the Weber's law)[13]. In the logarithmic domain, the smallest detectable change becomes $\Delta L = log(1 + \Delta I/I)$ which is constant for all intensity values. Assuming that Weber's law roughly holds for speech sounds as well, and that the speech spectrum power estimated at the ASR system front-end is proportional to the speech intensity, this means that, conceptually, all speech patterns that "sound" the same can be modeled with Gaussian functions with fixed variance equal to ΔL^2. Thus, any pattern whose log power spectrum lies farther than ΔL from any of the Gaussian means (that represent all patterns learned so far) can be considered a new, i.e. different, speech pattern. This makes ΔL suitable for a novelty detection criterion.

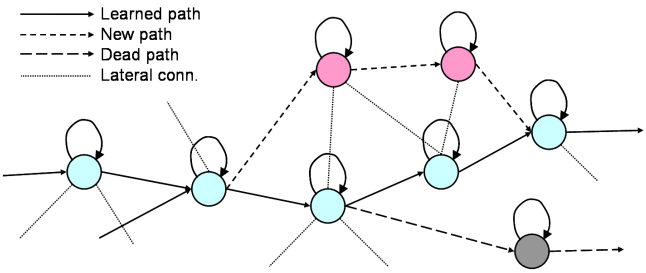

Figure 1: Schematic structure of a Dynamic Hidden Markov network.

Guided by the above consideration, we use a single multivariate Gaussian function with a fixed diagonal covariance matrix for the DHMnet state PDF and apply a threshold to the likelihood function for "novelty" detection. Since the DHMnet is a first-order Markov chain where input vectors are presumed conditionally independent, the pattern-level novelty detection can be substituted by multiple frame-level novelty detections. Thus, any given input vector x will be considered "new" if $P(x|\mu_b) < \theta$, where μ_b is the mean of the best matching state and the θ is the so-called *vigilance* threshold.

2.3. Stable learning

For the types of neural networks that we discussed in Section 1, the weights' update ΔW_n at each learning iteration is generally set to:

$$\Delta W_n = \alpha_n (X_n - W_{n-1}) \tag{1}$$

where X_n is the input vector and α_n is the learning rate at the n^{th} iteration. Stable learning is ensured when α_n is subject to the following constraints [10]:

$$\sum_{n=1}^{\infty} \alpha_n = \infty, \ \sum_{n=1}^{\infty} \alpha_n^2 < \infty \tag{2}$$

For the DHMnet state PDF learning, we consider the sequential version of the Maximum Likelihood estimation algorithm. In this case, the Gaussian mean update $\Delta\mu_n$ after input vector x_n will be:

$$\begin{aligned} \Delta\mu_n &= \mu_n - \mu_{n-1} = \frac{1}{n}\sum_{i=1}^{n} x_i - \frac{n}{n}\mu_{n-1} \\ &= \frac{(n-1)\mu_{n-1} + x_n - n\mu_{n-1}}{n} = \frac{1}{n}(x_n - \mu_{n-1}) \end{aligned} \tag{3}$$

which is exactly the same as Eq.(1). The learning rate is $\alpha_n = 1/n$ and it obviously satisfies the constraints of Eq.(2).

2.4. Topology representation

Since the DHMnet states represent different regions of the input feature space, it is important that neighboring states correspond to neighboring regions. That is, the state network should be a topology representing network. It has been shown that if lateral connections between neural network nodes (states in the DHMnet case) are built using the competitive Hebbian rule [7], the resulting network is a perfect topology representing network. The competitive Hebbian rule can be described as: for each input vector, connect the two closest nodes by an edge. Such networks have two very useful properties: 1) vectors that are neighbors in the input space will be represented by neighboring nodes; 2) if there is a path in the input space between two vectors, there will be a path connecting the two nodes that represent those vectors. These properties are often referred to as the neighborhood path preservation properties.

2.5. Removing "dead" states

When a network dynamically changes its structure, the state neighborhood relations also change. To account for these changes, each lateral connection is given an age that is set to zero when a connection is made or refreshed. Otherwise, the connection age is increased every time one of the connection's states is visited. This way, connections that reach a certain age, i.e. ones that have not been refreshed for some time, are removed. The DHMnet states can have many lateral connections and if for some state all connections are removed, this state is pronounced "dead" and is removed along with all transitions to and from it.

2.6. Time-Synchronous Decoding

For any input speech pattern represented by a sequence of feature vectors we are interested in finding the best state sequence or path through the network. Formally, this can be stated as follows:

$$\overline{S} = \max_{S} P(S|X), \quad X = \{x_i\}_i^T, \; S = \{s_i\}_1^T \tag{4}$$

The neighborhood and path preserving properties of the network ensure that each current state s_t is the best state given the current vector x_t. The best state sequence can be found by using a recursive procedure. Suppose that S_1^t is the best path until time t. Then

$$\begin{aligned} P(S_1^{t+1}|X_1^{t+1}) = & \qquad (5) \\ = & \max_{s_j \subset Succ(s_t)} P(s_j S_1^t | x_{t+1} X_1^t) \\ = & \max_{s_j \subset Succ(s_t)} P(s_j | S_1^t x_{t+1} X_1^t) P(S_1^t | x_{t+1} X_1^T) \\ = & \max_{s_j \subset Succ(s_t)} P(s_j | s_t x_{t+1}) P(S_1^t | X_1^t) \\ = & \left[\max_{s_j \subset Succ(s_t)} P(s_j|s_t) P(x_{t+1}|s_j) \right] P(S_1^t|X_1^t) \end{aligned}$$

where $Succ(s_t)$ is the set of succeeding states for state s_t, i.e. states that have incoming transitions from state s_t. This set includes s_t itself (because of the self-loop) and possibly a newly added state. The above recursion shows that the best state sequence can be obtained in a sequential time-synchronous manner by finding the best next state for each next input vector. Note that no backtracking is necessary as in the conventional Viterbi decoding algorithm.

2.7. Recognition with DHMnet

The DHMnet is designed to be the first processing block of a new kind of recognition system based entirely on the never-ending learning framework. Development of such system is currently underway, but is out of the scope of this paper. Here, we describe just the general principle of the DHMnet usage for recognition.

Recognition with DHMnet can be done by appropriate interpretation of the decoded best state sequence. In a similar manner to the way humans perform such task, paths through the network are associated with the characteristics of the patterns they represent. At first approximation, this means that each path and the corresponding states are labeled with all the information we had when this path was created or revisited. This can include the lexical content, speaker information, environment information, etc. When a speech utterance is presented to the network, in general, two cases can occur: 1) The decoded state sequence consists of only "old" states. This means that the whole speech pattern or all its segments have been already seen and learned. Then, we can recognize the input utterance from the path and state labels; 2) The decoded state sequence consists only or partly of newly added states. In this case, for each new state we can take the labels of its closest neighbor state and interpret the new states as "sounding like" their neighbors.

2.8. The DHMnet algorithm

We summarize the complete DHMnet algorithm as follows:

(1) Start with an empty network.

(2) For the next input vector x_t, given the current state s_{curr}, find the best matching succeeding state s_c. If it passes the vigilance test, set it as the next state, i.e. $s_{next} = s_c$, and go to (5).

(3) Find the best state, s_a, from all other states. If it passes the vigilance test, $s_{next} = s_a$, and go to (5).

(4) Add a new state, s_t, i.e. $s_{next} = s_t$, and set its mean to x_t.

(5) Make (update) the transition from the current state s_{curr} to s_{next}.

(6) Update the means of s_{next} and all its neighbors (Eq.3).

(7) Make (refresh) the connection between s_{next} and the second best state. Increase the ages of all s_{next} connections.

(8) If any connection age has reached the age threshold, remove this connection. Remove states with no connections.

(9) Add s_{next} to the best state sequence. Set the current state $s_{curr} = s_{next}$, and go to (2).

3. Experiments

For never-ending learning system such as the DHMnet, traditional evaluation schemes, i.e. dividing the available data into training, development and testing sets, model training, tuning and testing, do not make much sense. Furthermore, since this was the first time we experimented with a DHMnet, we were more interested in confirming that the network works as it is supposed to, than in obtaining recognition accuracy numbers.

For the experiments, we chose a small database of spelled letters utterances consisting of single samples of 22 English

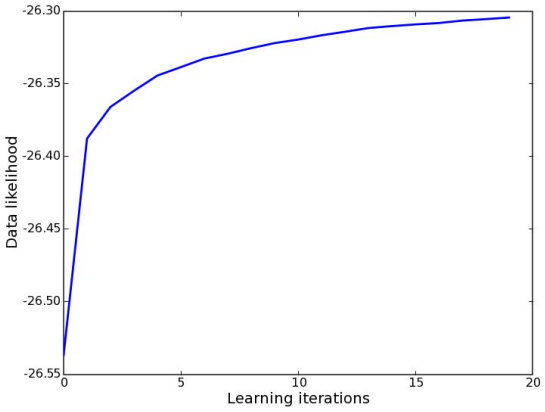

Figure 2: Likelihood change during 20 iterations of learning.

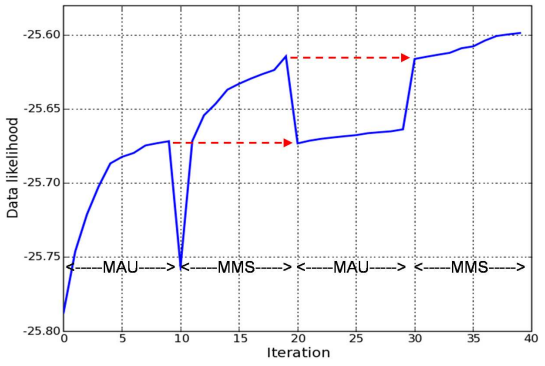

Figure 3: Likelihood change during alternating speaker data learning.

letters uttered by 20 (10 male and 10 female) Japanese speakers. In total, we had 440 utterances. Each utterance was transformed into a sequence of feature vectors consisting of 24 log filter-bank energies computed at a 10-ms. rate from 20-ms sliding windows. All DHMnet states' covariances were set to identity matrix and, respectively, the vigilance threshold was $\ln\theta = -12\ln(2\pi e)$.

In the first experiment, we tested the learning abilities of the network. Twenty learning iterations with all the data were performed (Fig.2 shows the observed data likelihood change). The increasing saturating curve clearly shows that the DHMnet is capable of stable learning. Next, to confirm that the network can learn new things without forgetting previously learned knowledge, we did the following experiment. First, we did 10 learning iterations with the data from only one speaker (MAU). Then, data from another speaker (MMS) were used for the next 10 iterations. After that, the data from MAU were given again to the network for another 10 iterations. Finally, the same procedure was repeated with MMS's data. Figure 3 shows the data likelihood during such learning. As can be seen, at the 20^{th} and 30^{th} iteration, when data changes to patterns that have been already seen, their likelihood continues increasing from the point where they were last seen. That means the learning with different speaker data did not destroy the previously stored knowledge, that is, the network can learn without catastrophic forgetting.

The last experiment was designed to check the recognition abilities of the network after each learning iteration. For each utterance, the decoded state sequence was stored and labeled with the speaker and letter ID. After each learning iteration, obtained state sequences were compared with those from the previous iteration to find the best matching sequence. If the labels matched, it was considered a hit. After only the second iteration, the recognition rate was 97.44%, and after the third and later iterations, it was 100%. Note that this means simultaneous speech and speaker recognition with no errors.

4. Conclusions

In this paper, we presented the Dynamic Hidden Markov network that in contrast to current speech models, is capable of never-ending unsupervised adaptive learning without catastrophic forgetting. We consider this network as a first processing block of a hierarchical system for full-scale speech recognition built according to the same learning principle. The DHMnet works in a single learning/decoding mode, but it can be easily extended with a pattern recall mode where sampling from the PDFs of the states along given path would reconstruct the corresponding speech pattern. A DHMnet that has these two modes of operation can be used not only for speech recognition, but also for speech synthesis, voice conversion, speech enhancement, etc.

5. References

[1] R. French, "Catastrophic Forgetting in Connectionist Networks," *Trends in Cognitive Sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[2] F. Hamker, "Life-long learning Cell Structures - continuously learning without catastrophic interference," *Neural Networks*, vol. 14, pp. 551–573, 2001.

[3] T. Kohonen, *Self-Organizaion and Associative Memory*, Springer-Verlag, Berlin, Germany, 3rd edition, 1989.

[4] S. Dusan and L. Rabiner, "On integrating insights from human speech perception into automatic speech recognition," in *Proc. Interspeech*, 2005, pp. 1233–1236.

[5] D. Hebb, *Organization of behavior*, Wiley, New York, 1949.

[6] S. Amari, "Field theory of self-organizing neural nets," *IEEE Trans. Syst. Man Cyber.*, vol. SMC-13, no. 5, pp. 741–748, 1983.

[7] T. Martinetz and K. Schulten, "Topology representing networks," *Neural Networks*, vol. 7, no. 3, pp. 507–522, 1994.

[8] S. Grossberg, "Nonlinear neural networks: principles, mechanisms, and architectures," *Neural Networks*, vol. 1, pp. 17–61, 1988.

[9] G. Carpenter and S. Grossberg, "The ART of adaptive pattern recognition by a self-organizing neural network," *Computer*, pp. 77–88, Mar. 1988.

[10] S. Furao and O. Hasegawa, "An incremental network for on-line unsupervised classification and topology learning," *Neural Networks*, vol. 19, pp. 90–106, 2006.

[11] N. Srinivasa and N. Ahuja, "A topological and temporal correlator network for spatiotemporal pattern learning, recognition and recall," *IEEE Trans. Neural Networks*, vol. 10, no. 2, pp. 356–371, Mar. 1999.

[12] D. Beroule, "An instance of coincidence detection architecture relying on temporal coding," *IEEE Trans. Nueral Networks*, vol. 15, no. 5, pp. 963–979, Sept. 2004.

[13] B. Moor, Ed., *Hearing*, Academic Press, 1995.