DOUBLY HIERARCHICAL DIRICHLET PROCESS HMM FOR ACOUSTIC MODELING

Amir Hossein Harati Nejad Torbati and Joseph Picone

Institute for Signal and Information Processing

College of Engineering, Temple University

Philadelphia, Pennsylvania, USA

{amir.harati, joseph.picone}@gmail.com

**Abstract**

Nonparametric Bayesian models use a Bayesian framework to learn model complexity automatically from the data and eliminate the need for a complex model selection process. A Hierarchical Dirichlet Process Hidden Markov Model (HDPHMM) is the nonparametric Bayesian equivalent of a hidden Markov model (HMM), but is restricted to an ergodic topology that uses a Dirichlet Process Model (DPM) to achieve a mixture distribution-like model. For applications involving ordered sequences (e.g., speech recognition), it is desirable to impose a left-to-right structure on the model. In this paper, we introduce a model based on HDPHMM that: (1) shares data points between states, (2) models non-ergodic structures, and (3) models non-emitting states. Previously a Doubly Hierarchical Dirichlet Process Hidden Markov Model (DHDPHMM) with a non-Ergodic structure has been introduced for supervised learning applications. Here we develop a generative extension of the DHDPHMM for semi-supervised applications. We present results of phoneme classification and recognition experiments on TIMIT. We show the proposed model produces a 20% relative reduction in error rate for phoneme classification and a 7% relative reduction on a speech recognition task.

*Index Terms*— nonparametric Bayesian models; HDP; HDPHMM, DHDPHMM; acoustic modeling

# Introduction

Hidden Markov models (HMMs) ‎[1]-‎[3] have been used successfully to model sequential data and have been applied to a wide range of applications including speech recognition. HMMs are parameterized both in their structure (e.g. number of states) and emission distributions (e.g. Gaussian mixtures). Model selection methods are traditionally used to optimize the number of states and mixture components. However, these methods are computationally expensive and there is no consensus on an optimum criterion for selection ‎[4].

Teh et al. ‎[5] and Fox et al. ‎[6] proposed an HMM with an unbounded number of states which is named HDPHMM since it is based on a hierarchical Dirichlet process (HDP) prior. HDPHMM is an ergodic model – a transition from an emitting state to all other states is allowed. This model has been used in speaker diarization ‎[6] and acoustic unit segmentation ‎[7]. However, in many pattern recognition applications involving temporal structure, such as speech processing, a left-to-right topology is required ‎[8]. Moreover, finite sequences often are modeled with initial and final non-emitting states ‎[8]. In the HDPHMM formulation, these problems are not addressed.

An HDPHMM uses an unbounded number of states and therefore data points are divided among many more states. Parameters for each state will be estimated with fewer data. Fox et al. ‎[6] introduced a model with state persistence that is biased to have fewer states but each state’s parameters are estimated independently of the other states. Sharing data, if performed carefully, can potentially improve the accuracy of the estimated parameters ‎[11]. Harati et al. ‎[9]‎[10] has proposed a model named Doubly Hierarchical Dirichlet Process Hidden Markov Model (DHDPHMM) that resolves these issues and allows mixture components to be shared amongst states. DHDPHMM introduces another parallel hierarchy to HDPHMM that enables sharing of mixture components. This is particularly important when modeling non-ergodic structures that tend to hypothesize more states than ergodic models.

In ‎[10], a DHDPHMM model with a non-ergodic structure has been introduced. An inference algorithm was also presented to learn the model from observations. It has been shown that DHDPHMM outperforms HDPHMM in problems similar to acoustic modeling and is computationally less expensive ‎[10]. However, the model introduced in ‎[10], as well as HDPHMM ‎[6], can only be used in supervised scenarios where word transcriptions along with time-alignments are available. In this paper we introduce a generative semi-supervised DHDPHMM and derive an approximation algorithm that can be used for seim-supervised training. We also provide phoneme classification and recognition results on the TIMIT dataset and show the proposed semi-supervised algorithm produces better results than completely supervised algorithms.

In Section ‎2 we present background material on nonparametric Bayesian modeling. In Section ‎3 DHDPHMM ‎[9]‎[10] is reviewed. In Section ‎4 we introduce a generative semi-supervised DHDPHMM. Finally in Section ‎5 experimental results are provided.

# Background

A Dirichlet process (DP) ‎[12] is a discrete distribution that consists of a countably infinite number of probability masses. A DP is denoted by DP(α,*H*), and is defined as ‎[12]:



where α is the concentration parameter, *H* is the base distribution ‎[12], and  is the unit impulse function at *θk*, often referred to as an atom ‎[13]. The weights *βk* are sampled through a stick-breaking construction ‎[14]:



The sequence of *βk* sampled by this process satisfies the constraint  with probability *1* and are denoted by *β~GEM(α)* ‎[14]. One of the main applications of a DP is to define a nonparametric prior distribution on the components of a mixture model. For example, a DP can be used to define a Gaussian mixture model (GMM) with an infinite number of mixture components ‎[13].

An HDP extends a DP to grouped data ‎[5]. In this case there are several related groups and the goal is to model each group using a mixture model. These models can be linked using traditional parameter sharing approaches. For example, consider the problem of modeling acoustic units (e.g., phonemes) using a mixture model in which parameters of the acoustic units can be shared. One approach is to use a DP to define a mixture model for each group and to use a global DP, DP(γ,*H*), as the common base distribution for all DPs ‎[5]. An HDP is defined as:



where *H* provides a prior distribution for the factor *θji*, *γ* governs the variability of *G0* around *H* and *α* controls the variability of *Gj*around *G0. H, γ* and *α* are hyperparameters of the HDP. We use a DP to define a mixture model for each group and use a global DP, DP*(γ,H)*, as the common base distribution for all DPs.

An HDPHMM ‎[6] is an HMM with an unbounded number of states. In a typical ergodic HMM, the number of states is fixed so that a matrix of dimension *N* states by *N* transitions per state is used to represent the transition probabilities. In an HDPHMM, the transition matrix is replaced by an infinite, but discrete, transition distribution, modeled by an HDP for each state. This lets each state have a different distribution for its transitions while the set of reachable states would be shared amongst all states.

Fox et al. ‎[6] extended the definition of HDPHMM to HMMs with state persistence by introducing a sticky parameter *κ*. The definition for HDPHMM is given by:



The state, mixture component and observation are represented by *zt*, *st* and *xt* respectively. The indices *j* and *k* are indices of the state and mixture components respectively. The base distribution that links all DPs together is represented by *β* and can be interpreted as the expected value of state transition distributions. The transition distribution for state *j* is a DP denoted by *πj* with a concentration parameter *α*. Another DP, *ψj*, with a concentration parameter *ϭ*, is used to model an infinite mixture model for each state (*zj*). The distribution *H* is the prior for the parameters *θkj*. If we want the posterior distribution over the parameters to remain in the same family as the prior, then *H* should be chosen to be a conjugate prior to the observation likelihood. Since the likelihood has a multivariate normal distribution, *H* should have a normal inverse Wishart (NIW) distribution.

# DHDPHMM with NON-ERGODIC STRUCTURE

We can extend the model in to address the problem of sharable mixture components. Equation defines a model with a multimodal distribution at each state. In HDPHMM these distributions are modeled using a DPM model:



Equation demonstrates when the state assignment, *zt*, for data point *xt* is known (or sampled previously), the mixture components can be sampled from a multinomial distribution with DP priors. Equation also shows that each emission distribution is modeled independent of other distributions. However, since in HDPHMM each state is also a group we can model the emission distribution using another HDP. The resulting model will have two parallel hierarchies and hence is referred to as a Doubly Hierarchical Dirichlet Process Hidden Markov Model (DHDPHMM). The generative model is defined as ‎[10]:



DHDPHMM pools the data points while HDPHMM divides data points between different states. If we don’t have enough data points in a particular state or a mixture component then the distribution parameters will be estimated poorly (e.g. the mean and covariance). For example, in speech recognition systems we usually use features with a dimensionality of *39* that translates to *820* free parameters per Gaussian mixture component assuming a full covariance (*39+(39x40)/2+1=820*). In practice we often assume a diagonal covariance matrix to reduce the parameter count.

A non-ergodic structure for DHDPHMM can be achieved by modifying the transition distributions. Here we only provide the extension to a left-to-right (LR) model but other types of models are also possible ‎[10].

The transition probability from state *j* has infinite support and can be written as:



Equation enforces an ergodic structure. In order to obtain a left-to-right (LR) topology we need to force the base distribution of the Dirichlet distribution in to only contain atoms to the right of the current state. This means *β* should be modified so that the probability of transiting to states left of the current state (i.e. states previously visited) becomes zero. For state *j* we define *Vj={Vji}:*



where *i* is the index for all following states. We can then modify *β* by multiplying it with *Vj*:



By replacing in we can obtain a generative model for left-to-right DHDPHMM. In ‎[9]‎[10] we have derived the inference algorithm for this model and provided two approaches to add non-emitting states to the model.

# A Semi-Supervised Generative model

Parametric HMMs have been trained in a semi-supervised setting using a straightforward extension of the Baum-Welch ‎[15] algorithm known as embedded training ‎[16]. A large HMM is composed for each utterance by connecting phoneme HMMs (including short pause) through their non-emitting states. The resulting composite HMM is then trained using the EM algorithm and this procedure is repeated for every utterance. However, this procedure cannot be applied directly to DHDPHMMs because a DHDPHMM learns its structure (e.g. number of states and the topology) from the corresponding observations. Therefore, there are no initial models available to generate a composite model. However, we can introduce a generative model that enables us to build a composite DHDPHMM for semi-supervised training.

Let’s assume we have a list of all models denoted by **m**:



where in this definition *ma* is the model corresponding to the phoneme */a/.* Let us also define an array of utterances, *U*. Each utterance is indexed by an integer number *u*. For example, consider the *27th* utterance in our list. For *u = 27*, we have: *U[27] = /hh ie s hh/.* Each utterance is a sequence of models defined in , and the model assignments for entire corpus are enumerated in *U*.

For each utterance we have an array, or list, of models. We can define a data structure, *Q*, which contains all models for all utterances, indexed by the position of the model in the list. For example, *Q[27]* = *{1:mhh, 2:mie, 3:ms, 4:mhh}*. Let *Lu* denote the total number of models used for utterance *u*:



We also need a function to return the model given the index of the model in the utterance, which we will define as *ϕ(i, u, Q)*:



In order to obtain a generative model for a composite DHDPHMM we need to develop a generative model for an utterance given the labels. Continuing our example of the *27th* utterance that consists of the models */hh ie s hh/*, we can represent these models with a Markov chain as shown in Fig. 1. Consider an observation sequence *Oj* and utterance membership function, *Φ*, with length *T*:





where *Ik* is the model index for the *kth* observation *ok*.

In order to generate *Oj* we first have to sample the Markov chain shown in Fig. 1 to determine which model produced the observation *ok*. We can write the boundary conditions as:



The Markov relationship can be written as:



where  and  is the self-transition probability for the model indexed by *Ii* in utterance *u*. Starting from *i = 1* we select the first model in the list and generate an observation using this model. We then use and to select the next model. This is either the same model (if a self-transition occurs) or the next model in the Markov chain.

A formal definition based on can be written as:



where in this equation each DHDPHMM is indexed by *ψ(It,u,Q)*. Equation defines a composite DHDPHMM for the utterance indexed by *u*. A composite DHDPHMM is defined for every utterance in the dataset. Therefore, to generate the entire dataset we have to first select an utterance index and then use this generative model to generate observations for that utterance.

It is possible to derive an inference algorithm for that directly samples DHDPHMM along with the segment variable, *It*. However, the resulting algorithm is computationally inefficient because we need to sample *It* which also requires sampling *ρ*. As a result computing the inference algorithm involves computing many likelihood functions and thousands of iterations before convergence. In the following we present an approximation algorithm to simulate a composite generative DHDPHMM.

**4.1. Approximation**

Exact inference of a composite DHDPHMM in is computationally difficult because all models are linked together through *It* and *xt*. However, if *It* was known, the composite DHDPHMM reduces to a collection of independent DHDPHMMs (*xt* can be segmented based on *It*). We can divide this problem into two sub-problems: (1) segmentation of the observations into aligned blocks with labels, and (2) DHDPHMM training. The first problem is a well-known problem in speech recognition and can be solved using a forced alignment process that is based on the Viterbi algorithm ‎[17]. The second problem has already been addressed in ‎[10].

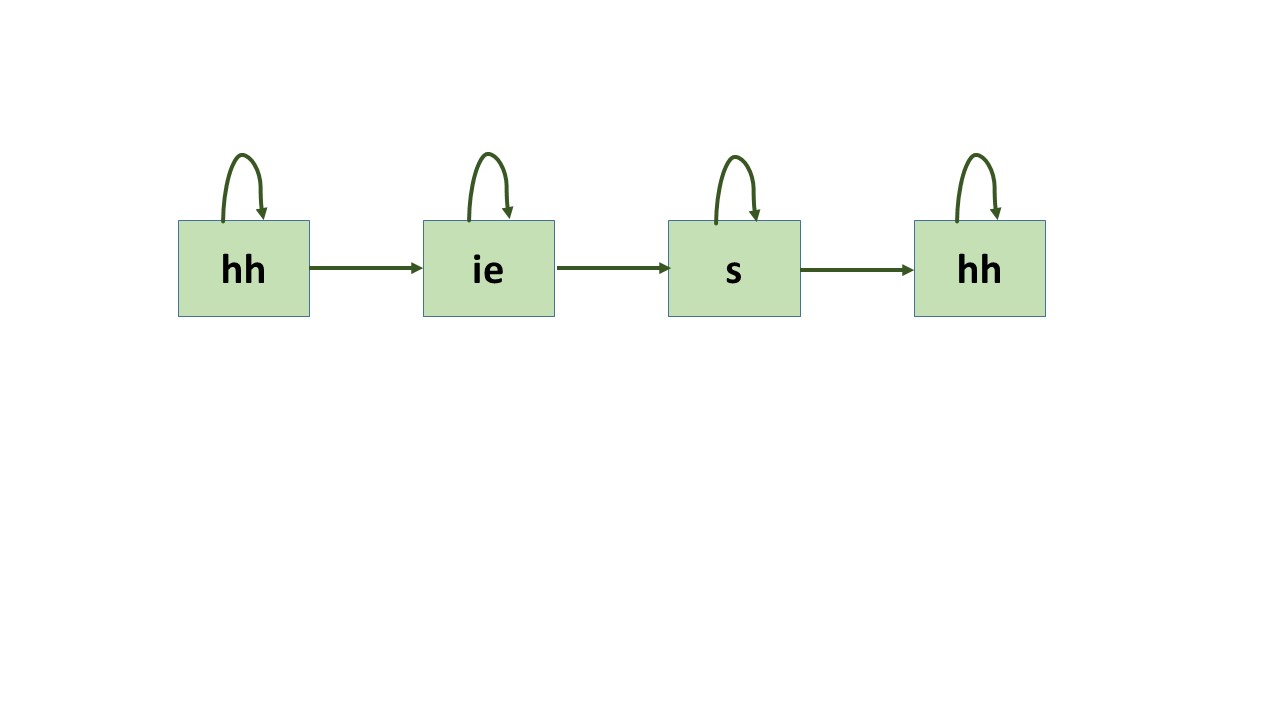


Fig. 1. A Markov chain representing a composite HMM

Assuming that we have a set of phoneme DHDPHMMs, we can generate a regular composite HMM by connecting these HMMs together based on a given sequence of labels and utilizing non-emitting states. Now we can use the Viterbi algorithm to find a time alignment between observations and states for the composite HMM. However, since these states can also be assigned to phoneme DHDPHMMs we can also find the alignment between the labels and observations. The algorithm is as follows:

1. Initialize the alignment using a heuristic method.

2. Use the alignment to generate a list of examples for each DHDPHMM.

3. Use examples generated in Step 2 to train each DHDPHMM.

4. Use models obtained in Step 3 to time align the labels to observations using the Viterbi algorithm (by first generating a composite HMM for each utterance).

5. If the maximum number of iterations is reached or a convergence happens, exit. Otherwise, return to Step 2.

In this algorithm, convergence can be checked by calculating the average log-likelihood of the models for training or a separate development data set can be used. Initialization can be done using any heuristic method.

# Experiments

In this section, some experimental results for phoneme classification and recognition will be presented. The TIMIT ‎[18] Corpus has been used for both classification and recognition tasks. The TIMIT Corpus ‎[18] is one of the most cited evaluation data sets used to compare new speech recognition algorithms. The data is segmented manually into phonemes and therefore is a natural choice to evaluate phoneme classification algorithms. TIMIT contains 630 speakers from eight main dialects of American English. There are a total of 6,300 utterances where 3,990 are used in the training set and 192 utterances are used for the “core” evaluation subset (another 400 used as development set). We followed the standard practice of building models for 48 phonemes and then map them into 39 phonemes ‎[19].

A standard 39-dimensional MFCC feature vector was used (12 Mel-frequency Cepstral Coefficients plus energy and their first and second derivatives) to convert speech data into feature streams. Cepstral mean subtraction for each utterance was also used.

**5.1. Phoneme Classification**

The first experiment is designed to compare DHDPHMM to other popular approaches for sub-word modeling in a supervised setting on a phoneme classification task. Table 1 shows a comparison between LR DHDPHMM and several other models including HDPHMM and discriminatively trained HMMs. As this table shows DHDPHMM is consistently better than their HDPHMM counterparts.

Further, it can be seen that LR models perform better than ergodic models. LR DHDPHMM finds *3,888* Gaussians for all *48* phonemes while two different LR HDPHMM models find *4,628* and *7,281* Gaussians for all phonemes respectively. These results show DHDPHMM can learn a less complex model that can explain the data better than a more complex model learned by HDPHMM. We can also see DHDPHMM works better than both ML and MMI trained HMMs with 5,760 Gaussians. Several other state of the art systems are shown that have error rates comparable to our model. However, these models are often trained discriminatively or use context while DHDPHMM is trained only using maximum likelihood and is context independent.

**5.2. Semi-Supervised Phoneme Recognition**

Table 1. Phoneme classification results

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Discr. | Dev.  % Err | Core % Err |
| HMM (40 Gauss.) | No | 25.0 | 26.1 |
| HMM/MMI (40 Gauss.) ‎[19] | Yes | 23.4 | 25.3 |
| HCRF/SGD ‎[19] | Yes | 20.3 | 21.7 |
| Large Margin GMMs ‎[20] | Yes | – | 21.1 |
| GMMs/Full Cov. ‎[20] | No | – | 26.0 |
| SVM ‎[21] | Yes | – | 22.4 |
| Data-driven HMM‎[22] | No | – | 21.4 |
| Direct Segmental Model ‎[23] | Yes | – | 21.7 |
| LR HDPHMM 1 | No | 23.5 | 24.4 |
| LR HDPHMM 2 | No | 23.8 | 25.1 |
| Ergodic DHDPHMM | No | 24.0 | 25.4 |
| LR DHDPHMM | No | 20.5 | 21.4 |

First, we investigate two different initialization techniques for the semi-supervised algorithm. Fig. 2 compares uniform initialization to a more traditional approach that uses forced alignment and CDHMMs. It can be seen that the forced alignment approach produced better results, though the differences diminish as more iterations of reestimation are used. From Fig. 2 it is evident that algorithm is sensitive to the initial segmentations. In the following we use the results of system initialized using forced alignments produced by a CDHMM system.

Table 2 presents results for DHDPHMM and several other models. The systems can be divided into two groups based on their training method (e.g., discriminative) and context modeling approach. We have also presented a comparison of supervised training of LR DHDPHMM with the semi-supervised algorithm. For the supervised case we have used the manual segmentations that are part of TIMIT. As this table shows, the semi-supervised algorithm outperforms its supervised counterpart (*29.02%* vs. *29.71%*). DHDPHMM performs better than a comparable CI HMM model. The error rate drops from *31.05%* for HMM to *29.02%* for CI DHDPHMM.

Table 2. Phoneme recognition results

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Discr. / Context | % Err | Set |
| Baseline CI-HMM | No/No | 31.05 | Core |
| Baseline CD-HMM | No/No | 27.93 | Core |
| CD-HMM 2 ‎[24] | No/Yes | 30.90 | Core |
| CD-HMM 3‎[11] | No/Yes | 27.70 | Core |
| HMM MMI 1‎[25] | Yes/No | 32.50 | Rand |
| HMM MMI 2 /Full Cov. ‎[25] | Yes/No | 30.30 | Rand |
| CD HMM /DM ‎[26] | Yes/Yes | 26.70 | – |
| HeterogeneousClass. ‎[27] | Yes/Yes | 24.40 | Core |
| Data-driven HMM ‎[22] | No/Yes | 26.40 | Core |
| Large Margin GMM ‎[20] | Yes/No | 30.10 | Core |
| CRF ‎[28] | Yes/No | 29.90 | All |
| Tandem HMM‎[28] | Yes/Yes | 30.60 | All |
| CNN/CRF‎[29] | Yes/Yes | 29.90 | Core |
| Direct Segmental Model‎[23] | Yes/No | 33.10 | Core |
| CIHCRF ‎[30]‎[30]MPE Init. | Yes/No | 28.30 | Core |
| CIHCRF ‎[30] ML Init. | Yes/No | 29.00 | Core |
| Deep CRF ‎[31] | Yes/Yes | 25.90 | Core |
| Deep Belief Network ‎[32] | Yes/Yes | 20.00 | Core |
| **CI LR DHDPHMM (Sup.)** | No/No | 29.71 | Core |
| **CI LR DHDPHMM (Sup.)** | No/No | 29.23 | All |
| **CI LR DHDPHMM (Semi.)** | No/No | 29.02 | Core |
| **CI LR DHDPHMM (Semi.)** | No/No | 28.36 | All |
| **CD LR DHDPHMM(Semi.)** | No/No | 27.51 | Core |
| **CD LR DHDPHMM(Semi.)** | No/No | 26.80 | All |

The third and fourth rows of Table 2 show two context-dependent HMMs. We can see that CI DHDPHMM performs slightly better than the CD model in row three (CD HMM 2) but slightly worse than CD model of row four (CD HMM 3). However, CD DHDPHMM works better than all CD HMM systems presented in this table (the difference is not statistically significant at significant level of *95%*). Also, our CI models are completely nonparametric Bayesian models while our CD DHDPHMMs are EM trained models based on CI DHDPHMM models. This is one of the reasons that the gain for CD models vanishes relative to the gain for CI models. Our CI model also performs better than a discriminatively trained CI HMM (MMI 1 and MMI 2) and its performance is comparable to discriminatively trained CI HCRF models. However, we can see that a discriminatively trained CD HMM (row seven) gives slightly better results relative to the CD DHDPHMM model trained only using maximum likelihood (but this difference is not statistically significant at significant level of *95%*).

Finally, it is also important to note that two of the models, Large Margin GMM and Direct Segmental Model ‎[23], which are state of the art for the phoneme classification task (see Table 1), perform much worse than our CI model.

# Conclusions

In this paper we have introduced a semi-supervised generative model for a left-to-right DHDPHMM. We have also derived an approximation algorithm to simulate this generative model. Through experimentation on TIMIT, we have shown that the proposed model can learn the optimum complexity from the data. This model outperforms both HMM (ML and MMI trained) and HDPHMM models.

One of the current problems with DHDPHMM and HDPHMM is that the inference algorithm is computationally expensive. It is a serious problem when we are dealing with large datasets typical in speech or video processing applications. Therefore, our next task is to improve the inference algorithm for LR DHDPHMM using its specific properties and structure.

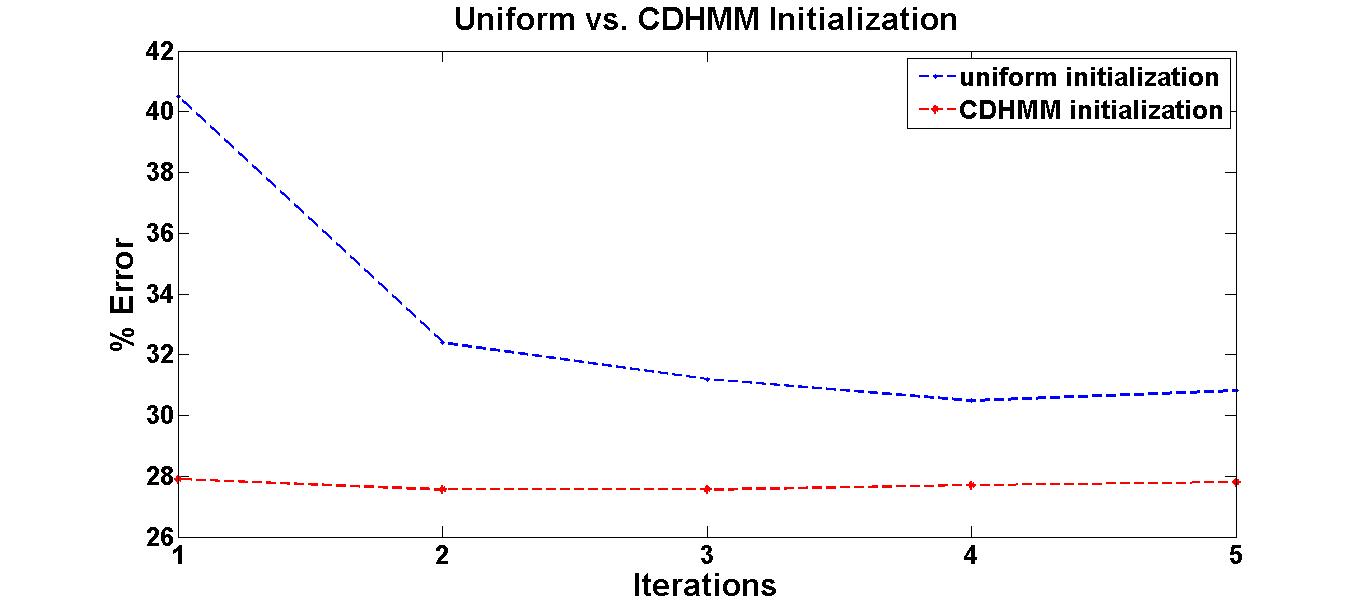


Fig. 2. A comparison of model initialization methods

# Acknowledgment

This research was supported in part by the National Science Foundation through Major Research Instrumentation Grant No. CNS-09-58854.

# References

1. L.. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition,” *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
2. B.-H. Juang and L. Rabiner, “Hidden Markov Models for Speech Recognition,” *Technometrics*, vol. 33, no. 3, pp. 251–272, 1991.
3. J. Picone, “Continuous Speech Recognition Using Hidden Markov Models,” *IEEE ASSP Magazine*, vol. 7, no. 3, pp. 26–41, Jul. 1990.
4. J. B. Kadane and N. A. Lazar, “Methods and Criteria for Model Selection,” *Journal of the American Statistical Association*, vol. 99, no. 465, pp. 279–290, 2004.
5. Y. Teh, M. Jordan, M. Beal, and D. Blei, “Hierarchical Dirichlet Processes,” *Journal of the American Statistical Association*, vol. 101, no. 47, pp. 1566–1581, 2006.
6. E. Fox, E. Sudderth, M. Jordan, and A. Willsky, “A Sticky HDP-HMM with Application to Speaker Diarization.,” *The Annalas of Applied Statistics*, vol. 5, no. 2A, pp. 1020–1056, 2011.
7. A. Harati, J. Picone, and M. Sobel, “Speech Segmentation Using Hierarchical Dirichlet Processes,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, p. TBD.
8. G. A. Fink, “Configuration of Hidden Markov Models From Theory to Applications,” in *Markov Models for Pattern Recognition*, Springer Berlin Heidelberg, 2008, pp. 127–136.
9. A. H. Harati Nejad Torbati, J. Picone, and M. Sobel, “A Left-to-Right HDP-HMM with HDPM Emissions,” in *Proceedings of the Conference on Information Sciences and Systems*, 2014, p. 1-6
10. A. Harati and J. Picone, “A Doubly Hierarchical Dirichlet Process Hidden Markov Model with a Non-Ergodic Structure,” submitted to *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2015 (available at: *http://www.isip.piconepress.com/publications/unpublished/journals/2015/dhdphmm/*).
11. S. Young and P. C. Woodland, “State Clustering in HMM-Based Continuous Speech Recognition,” *Computer Speech & Language*, vol. 8, no. 4, pp. 369–383, 1994.
12. Y.-W. Teh, “Dirichlet process,” in *Encyclopedia of Machine Learning*, Springer, 2010, pp. 280–287.
13. C. E. Rasmussen, “The Infinite Gaussian Mixture Model,” in *Proceedings of Advances in Neural Information Processing Systems*, 2000, pp. 554–560.
14. J. Sethuraman, “A constructive definition of Dirichlet priors,” *Statistica Sinica*, vol. 4, no. 2, pp. 639–650, 1994.
15. L. E. Baum and T. Petrie, “Statistical Inference for Probabilistic Functions of Finite State Markov Chains,” *The Annals of Mathematical Statistics*, vol. 37, no. 6, pp. 1554–1563, Dec. 1966.
16. S. Young, et al., *The HTK Book*, Cambridge, UK, 2006.
17. A. Viterbi, “Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm,” *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.
18. J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallet, N. Dahlgren, and V. Zue, “TIMIT Acoustic-Phonetic Continuous Speech Corpus,” *The Linguistic Data Consortium Catalog*, 1993.
19. A. Gunawardana, M. Mahajan, A. Acero, and J. C. Platt, “Hidden Conditional Random Fields for Phone Classification,” in *Proceedings of INTERSPEECH*, 2005, pp. 1117–1120.
20. F. Sha and L. K. Saul, “Large Margin Gaussian Mixture Modeling for Phonetic Classification and Recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2006, pp. 265–268.
21. P. Clarkson and P. J. Moreno, “On the Use of Support Vector Machines for Phonetic Classification,” in *IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP), 1999, pp. 585–588.
22. S. Petrov, A. Pauls, and D. Klein, “Learning Structured Models for Phone Recognition,” in *Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2007, pp. 897–905.
23. G. Zweig, “Classification and Recognition with Direct Segment Models,” in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, 2012, pp. 4161 – 4164.
24. K.-F. Lee and H.-W. Hon, “Speaker-Independent Phone Recognition Using Hidden Markov Models,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 11, pp. 1641–1648, 1989.
25. S. Kapadia, V. Valtchev, and S. Young, “MMI training for continuous phoneme recognition on the TIMIT database,” in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, 1993, pp. 491 – 494.
26. S. Watanabe, T. Hori, E. McDermott, and A. Nakamura, “A Discriminative Model for Continuous speech Recognition Based on Weighted Finite State Transducers,” in Proceeedings of the International Conference on Acoustics Speech and Signal Processing (ICASSP), 2010, pp. 4922 – 4925.
27. A. Halberstadt and J. R. Glass, “Heterogeneous acoustic measurements and multiple classifiers for speech recognition,” in *Ninth International Conference on Spoken Language Processing*, 1998, pp. 995–998.
28. J. Morris and E. Fosler-Lussier, “Conditional Random Fields for Integrating Local Discriminative Classifiers,” *IEEE Transactions on Acoustics, Speech and Language Processing*, vol. 16, no. 3, pp. 617 – 628, 2008.
29. D. Palaz, R. Collobert, and M. Magimai-Doss, “End-to-end Phoneme Sequence Recognition using Convolutional Neural Networks,” in *NIPS Deep Learning Workshop*, 2013.
30. Y.-H. Sung and D. Jurafsky, “Hidden Conditional Random Fields for Phone Recognition,” in *Proceedings of the IEEE Workshop on Speech Recognition and Understanding*, 2009, pp. 107 – 112.
31. D. Yu and L. Deng, “Deep-Structured Hidden Conditional Random Fields for Phonetic Recognition,” in *Proceedings of INTERSPEECH*, 2010, pp. 2986–2989.
32. G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohammed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, “Deep Neural Networks for Acoustic Modeling in Speech Recognition,” *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 83–97, Nov. 2012.