Hidden Markov Model Variational Autoencoder for Acoustic Unit Discovery

Janek Ebbers[1], *Jahn Heymann*[1], *Lukas Drude*[1], *Thomas Glarner*[1], *Reinhold Haeb-Umbach*[1], *Bhiksha Raj*[2]

[1]Paderborn University, Germany
[2]Carnegie Mellon University, United States

{ebbers,heymann,drude,glarner,haeb}@nt.upb.de, bhiksha@cs.cmu.edu

Abstract

Variational Autoencoders (VAEs) have been shown to provide efficient neural-network-based approximate Bayesian inference for observation models for which exact inference is intractable. Its extension, the so-called Structured VAE (SVAE) allows inference in the presence of both discrete and continuous latent variables. Inspired by this extension, we developed a VAE with Hidden Markov Models (HMMs) as latent models. We applied the resulting HMM-VAE to the task of acoustic unit discovery in a zero resource scenario. Starting from an initial model based on variational inference in an HMM with Gaussian Mixture Model (GMM) emission probabilities, the accuracy of the acoustic unit discovery could be significantly improved by the HMM-VAE. In doing so we were able to demonstrate for an unsupervised learning task what is well-known in the supervised learning case: Neural networks provide superior modeling power compared to GMMs.

Index Terms: variational autoencoder, hidden Markov Model, unsupervised learning, acoustic unit discovery

1. Introduction

Automatic Speech Recognition (ASR) performance has improved rapidly in the last years but is still highly dependent on the availability of large amounts of labeled training data. Not only is labeled data expensive, it may even be simply not available for rare languages. Therefore it is desirable to develop unsupervised learning algorithms which can make use of unlabeled data.

In a zero resource scenario where methods for speech processing are to be learnt from raw speech only, the tasks of acoustic and linguistic unit discovery can be discerned. While the former is concerned with finding phone-like subword units as acoustic building blocks of the language, the task of the latter is to discover semantically meaningful linguistic building blocks, i.e., word- or phrase-like units.

In this contribution we are concerned with Acoustic Unit Discovery (AUD). In [1] AUD is achieved by starting with a one-state HMM for all speech sounds, and modifying the successive state splitting algorithm [2] to successively learn the topology and parameters of HMMs to model the subword units. The authors of [3] carry out iterative re-estimation of the model parameters and unsupervised decoding to obtain the tentative label sequence. A sophisticated nonparametric Bayesian approach has been developed in [4], for which inference was carried out by Gibbs sampling, while inference in the full Bayesian model of [5] was achieved with the computationally more efficient variational inference.

In all these works GMMs were used as emission probabilities. This stands in contrast to the recent developments in supervised ASR, where neural networks have been shown to be superior to GMM-based acoustic modeling. While autoencoders have been successfully applied to unsupervised representation learning, see, e.g., [6], we attempt to employ variants of autoencoders for acoustic unit discovery.

Recently there has been a lot of research on generative neural networks. VAEs [7] allow learning of complex distributions and perform efficient inference by neural networks and have been successfully applied to unsupervised and semi-supervised learning of data point distributions [8, 9]. The Structured VAE (SVAE) proposed in [10] is a generalization of the VAE to more general graphical models, including those which capture the correlation structure of time signals. However, to the best of our knowledge the VAE and its extensions have not been applied to acoustic unit discovery yet.

Targeting unsupervised speech segmentation and acoustic model training in this paper, latent classes are introduced into the VAE model accounting for the acoustic units with temporal correlations being modeled by a conventional HMM. The resulting model enables the combination of well known HMMs with sophisticated emission distribution modeling by neural networks. All graphical model and neural network parameters are trained jointly by gradient-based optimization to maximize the likelihood of the generative model. The model structure is inspired by the work on SVAEs [10]. However, it is considerably simpler, adapted to the use with speech and due to the suggested training procedures it is more suitable to be applied with a large number of classes. Since the Viterbi algorithm [11] provides an efficient segmentation it can also be used for fast Viterbi training.

The paper is organized as follows. In the next section we recapitulate the concept of VAEs and then introduce the proposed HMM-VAE. We provide an overview of the inference and model estimation algorithms. Section 3 describes the AUD experiments we have conducted on the TIMIT database, while Section 4 offers some conclusions.

2. Variational Autoencoders

2.1. Generative Modeling

We aim to learn a generative model given a dataset $\mathcal{Y} = \{\mathbf{Y}_n\}$ composed of N independent observations $\mathbf{Y}_n$. An additional set of hidden variables $\mathcal{H} = \{\mathbf{H}_n\}$ is considered that allows us to model the data more precisely and/or to incorporate prior knowledge about the structure of the data that we are interested in, such as the acoustic units in AUD. The uppercase bold letters used here stand for arrays of vectors, i.e., for complete utterances. The marginal loglikelihood we would like to maximize with respect to its parameters θ is given by the sum of the

marginal loglikelihoods of the individual observations:

$$\ln p(\mathcal{Y}) = \sum_{n=1}^{N} \ln p(\mathbf{Y}_n). \tag{1}$$

Considering the given latent structure we can rewrite $p(\mathbf{Y}_n)$ as [7, 12]:

$$\ln p(\mathbf{Y}_n) = \mathrm{KL}\left(q(\mathbf{H}_n)||p(\mathbf{H}_n|\mathbf{Y}_n)\right) + \mathcal{L}\left(\mathbf{Y}_n\right), \tag{2}$$

where KL stands for the Kullback-Leibler divergence, and with

$$\begin{aligned}\mathcal{L}\left(\mathbf{Y}_n\right) &= \mathbb{E}_{q(\mathbf{H}_n)}\left[\ln p(\mathbf{Y}_n|\mathbf{H}_n)\right] + \mathbb{E}_{q(\mathbf{H}_n)}\left[\ln \frac{p(\mathbf{H}_n)}{q(\mathbf{H}_n)}\right] \\ &= \mathbb{E}_{q(\mathbf{H}_n)}\left[\ln p(\mathbf{Y}_n|\mathbf{H}_n)\right] - \mathrm{KL}\left(q(\mathbf{H}_n)||p(\mathbf{H}_n)\right).\end{aligned} \tag{3}$$

Optimization of Eq. (2) can be done by general Expectation-Maximization (EM) [12] by alternating between:

1. E-step: Inferring $q(\mathbf{H}_n)$ to approximate $p(\mathbf{H}_n|\mathbf{Y}_n)$.
2. M-step: Maximizing the lower bound $\mathcal{L}$ with respect to its parameters.

2.2. Variational Autoencoder

First we want to give a brief review of VAEs [7, 13], which constitute the basis of the model proposed in this paper. VAEs combine neural networks with variational inference to allow unsupervised learning of complicated distributions according to the graphical model shown in Fig. 1. Since single data points are

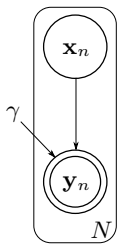

Figure 1: *Graphical Model considered with VAEs*

considered, bold lowercase letters are used to represent vectors here. A D_y-dimensional observation $\mathbf{y}_n$ is modeled in terms of a D_x-dimensional latent vector $\mathbf{x}_n$ using a non-linear transformation $f(\mathbf{x}_n;\gamma)$ with parameters γ:

$$\mathbf{y}_n = f(\mathbf{x}_n;\gamma) + \mathbf{v}_n, \tag{4}$$

with $\mathbf{v}_n \sim \mathcal{N}\left(\mathbf{0}, \sigma_y^2 \mathbf{I}_{D_y}\right)$ being Gaussian observation noise yielding the observation model:

$$p(\mathbf{y}_n|\mathbf{x}_n;\gamma) = \mathcal{N}\big(\mathbf{y}_n; f(\mathbf{x}_n;\gamma), \sigma_y^2 \mathbf{I}_{D_y}\big), \tag{5}$$

where $\mathbf{I}_{D_y}$ is the D_y-dimensional identity matrix. The transformation $f(\mathbf{x};\gamma)$ is given by a neural network which is referred to as probabilistic decoder as it provides the mean of our observation model. The vector $\mathbf{x}_n$ can be seen as code of the corresponding observation, which is assumed to be drawn from a standard Normal distribution $\mathbf{x}_n \sim p(\mathbf{x}_n) = \mathcal{N}\left(\mathbf{x}_n; \mathbf{0}, \mathbf{I}_{D_x}\right)$. The choice of this distribution is justified due to the fact that it can be transformed to an arbitrary distribution given a sufficiently complex transformation $f(\mathbf{x}_n;\gamma)$. Note that the noise variance σ_y^2, which is assumed constant here, could also be modeled dependent on $\mathbf{x}_n$ as a second output of the decoder network.

The VAE can be understood as a non-linear version of factor analysis [12].

2.2.1. E-step: Inference

Aiming at maximizing the marginal loglikelihood in Eq. (2), however, exact inference of the posterior $p(\mathbf{x}_n|\mathbf{y}_n)$ is not tractable because of the non-linear transformation. The idea of the VAE is to perform variational inference

$$q(\mathbf{x}_n;\phi) = \mathcal{N}\big(\mathbf{x}_n; \boldsymbol{\mu}_n, \mathrm{diag}(\boldsymbol{\sigma}_n^2)\big) \tag{6}$$

by another neural network with parameters ϕ, which is referred to as probabilistic encoder as it provides means and variances of the approximate posterior:

$$\big(\boldsymbol{\mu}_n, \ln \boldsymbol{\sigma}_n^2\big) = g(\mathbf{y}_n;\phi). \tag{7}$$

The encoder and decoder neural networks are trained jointly and hence inference is basically learned during training.

2.2.2. M-step: Maximization of the Objective Function

The objective function for gradient-based optimization of both the encoder and the decoder networks is given by the lower bound

$$\begin{aligned}\mathcal{L}\left(\mathbf{y}_n;\gamma,\phi\right) = \frac{1}{L}\sum_{l=1}^{L} \ln p(\mathbf{y}_n|\tilde{\mathbf{x}}_n^{(l)};\gamma) \\ - \mathrm{KL}\left(q(\mathbf{x}_n;\phi)||p(\mathbf{x}_n)\right),\end{aligned} \tag{8}$$

which corresponds to Eq. (3) with $\mathbf{x}_n$ denoting the latent variable $\mathbf{H}_n$ and with the first term of the right hand side of Eq. (3) approximated by sampling. We can backpropagate through the Gaussian sampling by using the reparameterization trick [7] $\tilde{\mathbf{x}}_n^{(l)} = \boldsymbol{\sigma}_n \odot \boldsymbol{\epsilon}_n^{(l)} + \boldsymbol{\mu}_n$ with standard Normally distributed samples $\boldsymbol{\epsilon}_n^{(l)}$. All terms in Eq. (8) can be calculated and differentiated in closed form.

2.3. HMM-VAE

We would like to combine VAEs with HMMs to model speech utterances leveraging complex emission distributions. Inspired by the SVAE proposed in [10] we therefore extend the VAE by latent states and model temporal correlations by HMMs. Each of the U acoustic units is modeled by three states with the typical left-to-right topology resulting in $K{=}3U$ states. The latent standard Normal distribution of a VAE is replaced by a state specific Normal distribution:

$$p(\mathbf{x}_{n,t}|z_{n,t}=k;\theta) = \mathcal{N}\big(\mathbf{x}_{n,t}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\big), \tag{9}$$

where θ denotes the set of graphical model parameters. The resulting generative process is illustrated in Fig. 2, where n, t denote the utterance index and the frame index within an utterance, respectively. In the following we will drop n and consider an individual utterance of length T with its observations $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_T]$ and latent variables $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_T]$ and $\mathbf{Z} = [z_1, ..., z_T]$ with

$$\ln p(\mathbf{Y}|\mathbf{X};\gamma) = \sum_{t=1}^{T} \ln p(\mathbf{y}_t|\mathbf{x}_t;\gamma), \tag{10}$$

$$\ln p(\mathbf{X}|\mathbf{Z};\theta) = \sum_{t=1}^{T} \ln p(\mathbf{x}_t|z_t;\theta), \tag{11}$$

$$\ln p(\mathbf{Z};\theta) = \sum_{t=1}^{T} \ln p(z_t|z_{t-1};\theta). \tag{12}$$

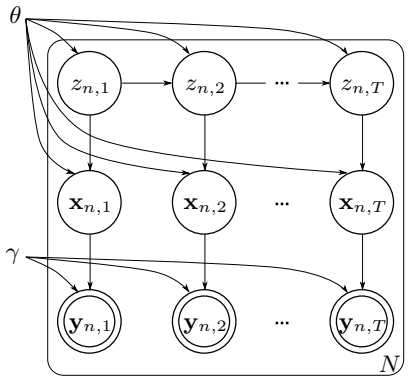

Figure 2: *Graphical Model of an HMM-VAE.*

The transition probabilities are given by

$$p(z_t = j|z_{t-1} = k; \theta) = a_{kj}, \tag{13}$$

where, in a slight abuse of notation, the term $p(z_1 = k|z_0; \theta)$ in Eq. (13) has to be understood to denote the initial state probability $p(z_1 = k; \theta) = \pi_k$ to simplify notation. In order to ensure that estimated probabilities sum up to one and that covariance matrices are symmetric we choose the graphical model parameters to be $\theta = \left\{ \ln \boldsymbol{\rho}, \{\ln \mathbf{r}_k, \boldsymbol{\mu}_k, \boldsymbol{C}_k\}_{k=1}^K \right\}$, with $\boldsymbol{\pi} = \mathrm{softmax}(\ln \boldsymbol{\rho})$ and transition probabilities, means and covariances are given by $\mathbf{a}_k = \mathrm{softmax}(\ln \mathbf{r}_k)$, $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k = \mathbf{C}_k \mathbf{C}_k^{\mathrm{T}} + \sigma_{\min}^2 \mathbf{I}_{D_x}$, respectively, for each state k.

As a VAE can be understood as a non-linear factor analysis, the proposed model can be considered as a non-linear version of a factor analysed HMM [14].

2.3.1. E-step: Inference

Using the following mean field approximation:

$$q(\mathbf{X}, \mathbf{Z}; \theta, \phi) = q(\mathbf{Z}; \theta, \phi) \prod_{t=1}^{T} q(\mathbf{x}_t; \phi) \tag{14}$$

with $q(\mathbf{x}_t; \phi)$ being provided by a probabilistic encoder network as before, we can perform mean field inference [12] to obtain

$$\begin{aligned} &\ln q(\mathbf{Z}; \theta, \phi) \\ &= \mathbb{E}_{q(\mathbf{X};\phi)} \left[\ln p(\mathbf{X}|\mathbf{Z}; \theta) \right] + \ln p(\mathbf{Z}; \theta) + \text{const.} \\ &= \sum_{t=1}^{T} \Big(b(z_t) + \ln p(z_t|z_{t-1}; \theta) \Big) + \text{const.} \end{aligned} \tag{15}$$

where const. is a normalizing constant and

$$b(z_t) = \mathbb{E}_{q(\mathbf{x}_t;\phi)} \left[\ln p(\mathbf{x}_t|z_t; \theta) \right]. \tag{16}$$

To avoid backpropagation through time during training we choose $\tilde{q}(\mathbf{Z}) := q(\mathbf{Z}; \theta^{(\text{old})}, \phi^{(\text{old})})$. Thus the dependency of $q(\mathbf{Z})$ on the parameters is not taken into account during differentiation of the final objective in the M-Step, which is the common approach in conventional EM. The Viterbi algorithm [11] provides an efficient way to find the most probable state sequence. Note that inference and hence segmentation can be performed without the use of the probabilistic decoder $f(\mathbf{x}_n; \gamma)$.

2.3.2. M-step: Maximization of Objective Function

Based on Eq. (3) with latent variables $\mathbf{H}_n = (\mathbf{X}_n, \mathbf{Z}_n)$ we obtain the objective function:

$$\mathcal{L}(\mathbf{Y}; \gamma, \theta, \phi) = \sum_{t=1}^{T} \mathcal{L}(\mathbf{y}_t; \gamma, \theta, \phi) \tag{17}$$

with

$$\begin{aligned} \mathcal{L}(\mathbf{y}_t; \gamma, \theta, \phi) = &\frac{1}{L} \sum_{l=1}^{L} \ln p(\mathbf{y}_t|\tilde{\mathbf{x}}_t^{(l)}; \gamma) \\ &- \mathbb{E}_{\tilde{q}(z_t)} \left[\mathrm{KL}\left(q(\mathbf{x}_t; \phi) || p(\mathbf{x}_t|z_t; \theta) \right) \right] \\ &+ \mathbb{E}_{\tilde{q}(z_{t-1}, z_t)} \left[\ln p(z_t|z_{t-1}; \theta) \right] \\ &+ \text{const.} \end{aligned} \tag{18}$$

where const. gathers all terms that are irrelevant for the optimization. The marginals $\tilde{q}(z_t)$ and $\tilde{q}(z_{t-1}, z_t)$ can be computed recursively using the Forward-Backward (FB) algorithm. However, as calculation of the expectation in the second term might be computationally expensive when marginalizing over a large number of states, it is reasonable to either perform Viterbi training or to approximate the expectation by sampling:

$$\begin{aligned} &\mathbb{E}_{\tilde{q}(z_t)} \left[\mathrm{KL}\left(q(\mathbf{x}_t; \phi) || p(\mathbf{x}_t|z_t; \theta) \right) \right] \\ &\approx \frac{1}{J} \sum_{j=1}^{J} \mathrm{KL}\left(q(\mathbf{x}_t; \phi) || p(\mathbf{x}_t|z_t^{(j)}; \theta) \right) \end{aligned} \tag{19}$$

with $z_t^{(j)} \sim \tilde{q}(z_t)$.

Note that an objective $\mathcal{L}(\mathbf{y}_t, z_t; \gamma, \theta, \phi)$ for a supervised scenario, where the state sequence is given, can be easily obtained from Eq. (18) by removing the expectations of the two latter terms. This is another attractive feature of the proposed HMM-VAE: It can be used for both supervised and unsupervised training, as well as for semi-supervised training, where some utterances come with labels while others do not.

The negative of the objective in Eq. (18) provides a loss function consisting of three loss terms. Notice that the first term becomes a sum over negative loglikelihoods which is basically a Mean Squared Error (MSE) between the output of the probabilistic decoder $f(\mathbf{x}_t^{(l)}; \gamma)$ and the actual observation $\mathbf{y}_t$:

$$-\ln p(\mathbf{y}_t|\tilde{\mathbf{x}}_t^{(l)}; \gamma) = \frac{||f(\mathbf{x}_t^{(l)}; \gamma) - \mathbf{y}_t||^2}{2\sigma_y^2} + \text{const.} \tag{20}$$

Hence it represents a reconstruction loss that can be scaled by the hyperparameter σ_y^2. The two latter terms of the loss function can be seen as regularization encouraging the model to learn a latent representation according to the incorporated model structure.

The overall loss function given as a sum over all utterances n can be minimized using minibatch training and gradient descent with respect to all parameters.

3. Experiments

For evaluation the proposed model is used for unsupervised learning of acoustic units on the TIMIT database [15]. Training and testing is performed on the complete datasets including the dialect sentences (SA), where 100 randomly chosen training utterances are used for cross validation. Mel Frequency Cepstral

Table 1: *Performance of different training types. The first column states the type of the initial segmentation used for pre-training (see text). The terms in parentheses state the results without training of the HMM parameters. The first row is the performance of the GMM-HMM system.*

Init.	Train.-Alg.	NMI	eq. PER
1	-	37.8%	65.4%
1	Viterbi	42.2%	58.3%
1	FB	42.3%	60.5%
2(a)	Viterbi	42.8% (42.8%)	58.9% (59.7%)
2(a)	FB	42.6%	59.0%
2(b)	Viterbi	41.9%	59.7%
2(b)	FB	42.5%	58.8%

Coefficents (MFCCs) are used as features with delta and delta-deltas where mean and variance were normalized to zero and one, respectively, for each feature and utterance. Each acoustic unit is modeled by an HMM with three states in a left-to-right topology. All transition probabilities within and between acoustic units are trained jointly.

Similar as in the hybrid approach of supervised ASR training, our HMM-VAE is initialized using the segmentation of a GMM-HMM system: First, variational Bayesian inference in a GMM-HMM model for AUD is carried out according to [5]. Then, to initialize our model we perform a pre-training, where the result of the GMM-HMM system is considered as the tentative supervised segmentation. To be specific, we compared three types of initializations:

1. Using the exact phone-level segmentation provided by the GMM-HMM system and deriving a state-level alignment by assuming equal duration of all states within a segment.
2. Using only the state-level label sequence, for each utterance, found by the GMM-HMM system and discarding the segment boundary information and either
   (a) deriving a state-level alignment by assuming equal duration of all states within an utterance or
   (b) using the given state-level label sequence as constraint in the Viterbi/FB algorithm.

Finally, the subsequent training is performed completely unsupervised. Note that the number U of acoustic units to be learned is given by the GMM-HMM system, which employs a Dirichlet process prior on the HMM inventory. Here U=72 acoustic units were discovered by the GMM-HMM. This number is not changed during the HMM-VAE training unless an HMM does not take responsibility for any of the observations, which would result in removal of that model.

The encoder and decoder networks consist of 2×256 hidden units each. Adam [16] is used for optimization with $\alpha=10^{-3}$. In all experiments L=1 sample is used to approximate the expected reconstruction loss. When not using Viterbi training, the second term of our objective is approximated by J=3 samples according to Eq. (19). Both, pre- and unsupervised training are terminated after three epochs without improvement of the cross validation loss according to Eq. (18).

The performance of the HMM-VAE is evaluated in terms of the Normalized Mutual Information (NMI) [17] and an equivalent Phone Error Rate (PER) between the learned segmentation and the true phone labeling. The NMI, defined as $\text{NMI} = \frac{I(\mathbf{Z}^{(\text{true})};\mathbf{Z}^{(\text{pred})})}{H(\mathbf{Z}^{(\text{true})})} = \frac{H(\mathbf{Z}^{(\text{true})})-H(\mathbf{Z}^{(\text{true})}|\mathbf{Z}^{(\text{pred})})}{H(\mathbf{Z}^{(\text{true})})}$ with $H(\cdot)$ representing an entropy, can be understood as a measure of statistic dependence between the predicted and the true label sequence with a high NMI being desirable. For the calculation of an equivalent PER the true phone labels were mapped from 61 to 39 classes as proposed by [18] and each acoustic unit was, similar to the calculation of the many-to-one Word Error Rate (WER) in [19], mapped to the ground truth phone with which it overlaps the most. The PER is then given by $\text{PER} = \frac{\text{Sub.}+\text{Del.}+\text{Ins.}}{\text{Tot.}}$ with Sub. + Del. + Ins. being the minimal number of substitutions, deletions, and insertions between the predicted and true phone sequence and Tot. the total number of phones in the true phone sequence with a low PER being desirable.

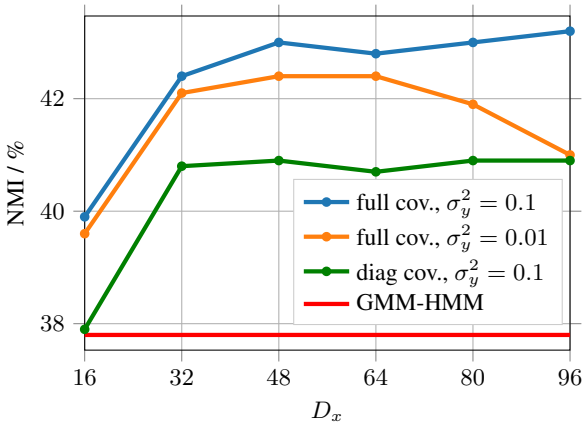

Figure 3: *NMI performance for different code dimensions D_x, types of covariance matrices and observation variances σ_y^2.*

Tab. 1 compares the different types of initialization and training algorithms using a latent code of dimension D_x=64, full covariance matrices and an observation variance of σ_y^2=0.1. We can see that in all cases the performance of the GMM-HMM based AUD training is significantly improved. However, the type of initialization and the used training algorithm only have a slight impact on the results. Note, however, that Viterbi training significantly reduces training time.

Fig. 3 shows the NMI for different code dimensions, types of covariance matrices and observation variances when using an initialization according to 2(a) and Viterbi training.

4. Conclusions

Inspired by the concept of structured VAEs [10] we have extended the VAE to capture temporal correlations in a speech signal by incorporating the structure of an HMM. An iterative EM-like algorithm for optimization of the objective function and inference of the latent variables has been derived. The concept was applied to an unsupervised acoustic unit discovery task, resulting in a significantly improved accuracy of the discovered acoustic units, compared to a variational Bayesian GMM-HMM algorithm, whose result was used as initialization for the proposed HMM-VAE. It is important to note that the concept is rather general and applicable to structures other than HMMs, although HMMs are the most prominent graphical model structure used in speech.

5. Acknowledgements

This work was in part supported by Deutsche Forschungsgemeinschaft (DFG) under contract no Ha3455/12-1. We thank Lucas Ondel and the whole Brno speech team for providing us the code for the variational inference based AUD training [5].

6. References

[1] B. Varadarajan, S. Khudanpur, and E. Dupoux, "Unsupervised Learning of Acoustic Sub-word Units," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, Stroudsburg, PA, USA, 2008.

[2] J. Takami and S. Sagayama, "A successive state splitting algorithm for efficient allophone modeling," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1992.

[3] M.-H. Siu, H. Gish, A. Chan, W. Belfield, and S. Lowe, "Unsupervised Training of an HMM-based Self-organizing Unit Recognizer with Applications to Topic Classification and Keyword Discovery," *Computer Speech and Language*, vol. 28, no. 1, pp. 210–223, 2014.

[4] C.-Y. Lee and J. Glass, "A Nonparametric Bayesian Approach to Acoustic Model Discovery," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers*, 2012.

[5] L. Ondel, L. Burget, and J. Cernocky, "Variational Inference for Acoustic Unit Discovery," *Procedia Computer Science*, vol. 81, pp. 80 – 86, 2016, sLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages 09-12 May 2016 Yogyakarta, Indonesia.

[6] D. Renshaw, H. Kamper, A. Jansen, and S. Goldwater, "A comparison of neural network methods for unsupervised representation learning on the zero resource speech challenge," in *INTERSPEECH*, 2015, pp. 3199–3203.

[7] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *ArXiv e-prints*, vol. abs/1312.6114, 2013.

[8] N. Dilokthanakul, P. A. M. Mediano, M. Garnelo, M. C. H. Lee, H. Salimbeni, K. Arulkumaran, and M. Shanahan, "Deep Unsupervised Clustering with Gaussian Mixture Variational Autoencoders," *CoRR*, vol. abs/1611.02648, 2016.

[9] D. P. Kingma, D. J. Rezende, S. Mohamed, and M. Welling, "Semi-Supervised Learning with Deep Generative Models," *CoRR*, vol. abs/1406.5298, 2014.

[10] M. Johnson, D. K. Duvenaud, A. Wiltschko, R. P. Adams, and S. R. Datta, "Composing graphical models with neural networks for structured representations and fast inference," in *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2016, pp. 2946–2954.

[11] G. D. Forney, "The Viterbi algorithm," *Proc. of the IEEE*, vol. 61, pp. 268 – 278, March 1973.

[12] C. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[13] C. Doersch, "Tutorial on Variational Autoencoders," *ArXiv e-prints*, vol. abs/1606.05908, 2016.

[14] A. I. Rosti and M. Gales, "Factor analysed hidden markov models for speech recognition," *Computer Speech & Language*, vol. 18, no. 2, pp. 181–200, 2004.

[15] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus CDROM," 1993.

[16] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *ArXiv e-prints*, 2014.

[17] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

[18] K. F. Lee and H. W. Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1989.

[19] H. Kamper, A. Jansen, and S. Goldwater, "A segmental framework for fully-unsupervised large-vocabulary speech recognition," *CoRR*, vol. abs/1606.06950, 2016.