Interspeech 2018
2-6 September 2018, Hyderabad

Full Bayesian Hidden Markov Model Variational Autoencoder for Acoustic Unit Discovery

Thomas Glarner, Patrick Hanebrink, Janek Ebbers, Reinhold Haeb-Umbach

Paderborn University, Germany
{glarner,ebbers,haeb}@nt.upb.de

Abstract

The invention of the Variational Autoencoder enables the application of Neural Networks to a wide range of tasks in unsupervised learning, including the field of Acoustic Unit Discovery (AUD). The recently proposed Hidden Markov Model Variational Autoencoder (HMMVAE) allows a joint training of a neural network based feature extractor and a structured prior for the latent space given by a Hidden Markov Model. It has been shown that the HMMVAE significantly outperforms pure GMM-HMM based systems on the AUD task. However, the HMMVAE cannot autonomously infer the number of acoustic units and thus relies on the GMM-HMM system for initialization. This paper introduces the Bayesian Hidden Markov Model Variational Autoencoder (BHMMVAE) which solves these issues by embedding the HMMVAE in a Bayesian framework with a Dirichlet Process Prior for the distribution of the acoustic units, and diagonal or full-covariance Gaussians as emission distributions. Experiments on TIMIT and Xitsonga show that the BHMMVAE is able to autonomously infer a reasonable number of acoustic units, can be initialized without supervision by a GMM-HMM system, achieves computationally efficient stochastic variational inference by using natural gradient descent, and, additionally, improves the AUD performance over the HMMVAE.

Index Terms: Acoustic Unit Discovery, Bayesian, Structured Variational Autoencoders, Underresourced Languages

1. Introduction

The task of acoustic unit discovery (AUD) can be defined as segmenting speech while simultaneously clustering these segments into a reasonable number of acoustic units without the help of supervision through training labels. As such, it can be seen as the unsupervised counterpart of acoustic modeling known from automatic speech recognition. One major application domain is the treatment of underresourced languages, where annotated databases and linguistic resources are scarce.

Since the introduction of the variational autoencoder (VAE) by [1], various attempts have been made to extend this type of deep generative modeling into different directions, with AUD among many others. Especially the structured VAE (SVAE) framework developed by [2] is well suited for AUD since it offers the advantage of combining a traditional probabilistic graphical model (PGM) with the power of deep neural networks (NNs). A current model proposed in [3], called the hidden Markov model variational autoencoder (HMMVAE), is an instance of the SVAE [2] specifically designed to tackle the task of generative acoustic modeling and AUD. Here, one hidden Markov model (HMM) is used for every hypothesized acoustic unit to structure the latent space. Experiments have shown promising improvements when compared to the widely known GMM-HMM AUD system proposed in [4]. In this work, we aim to improve the HMMVAE by extending it with a full Bayesian treatment of the latent probabilistic model. Most importantly, this extension also allows the model to autonomously learn the necessary number of acoustic units by making use of the Dirichlet process (DP) known from Bayesian nonparametrics [5]. In addition, to impose a regularization effect on the PGM learning process, all of its learned parameters now have appropriate prior distributions. However, the encoder and decoder NNs of the model are learned by a minibatch stochastic gradient descent (SGD) based algorithm. Therefore, utilizing the usual variational expectation-maximization (EM) algorithm with a graphical model parameter update only after every epoch would lead to mismatched learning velocities between NNs and PGM and thus slow convergence. We side-step this by employing stochastic variational inference (SVI) to train the PGM as proposed in [6]. This requires a reparametrization of the PGM in terms of exponential family distributions.

2. Model Description

2.1. Variational Autoencoder

The standard VAE is a generative model able to represent a complex data distribution on the observation $\mathbf{y}$ by combining simple distributions and neural networks. In this section, all underlying distributions are Gaussians and the data is assumed to be independent and identically distributed (i.i.d.). The assumption that an underlying latent code vector $\mathbf{x}$ has caused the observation can be stated as a conditional density called decoder:

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; f(\mathbf{x};\delta), \sigma^2\mathbf{I}), \qquad (1)$$

where the function $f(\mathbf{x};\delta)$ is a NN paremetrized by the set δ to capture complex relationships between the observation and the code, while the latter is assumed to be drawn from a zero-mean, isotropic unit-variance Gaussian prior distribution: $p(\mathbf{x}) = \mathcal{N}(\mathbf{x};\mathbf{0},\mathbf{I})$. Since the true posterior $p(\mathbf{x}|\mathbf{y})$ is intractable due to the NN, an approximated posterior, called the encoder, is postulated as a Gaussian with mean and variance vectors depending on the observation,

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

where the mapping is given by yet another NN parameterized by ϕ: $(\boldsymbol{\mu}_n, \boldsymbol{\sigma}_n) = g(\mathbf{y}_n;\phi)$. Given the dataset $\mathcal{Y} = \{\mathbf{y}_1,\ldots,\mathbf{y}_N\}$ and assuming a corresponding set of latent codes $\mathcal{X} = \{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$, the parameters of both NNs can be learned by maximizing the evidence lower bound (ELBO) – or, equivalently, minimizing its negative – which decomposes into terms

10.21437/Interspeech.2018-2148

involving only one observation each:

$$\log p(\mathcal{Y}) \geq \mathcal{L}^{(\text{VAE})}(\delta,\phi) = \mathbb{E}_{q(\mathcal{X})}\left[\log \frac{p(\mathcal{Y},\mathcal{X};\delta)}{q(\mathcal{X};\phi)}\right]$$
$$= -\sum_{n=1}^{N} \frac{\mathbb{E}_{q(\mathbf{x}_n)}\|\mathbf{y}_n - f(\mathbf{x}_n;\delta)\|^2}{2\sigma^2} - \mathbb{E}_{q(\mathbf{x}_n)}\left[\log\frac{p(\mathbf{x}_n)}{q(\mathbf{x}_n;\phi)}\right].$$

The first term, usually called the reconstruction loss, has no closed form due to the complicated dependence on $\mathbf{x}_n$. This is resolved through a sampling approximation of the expectation while making use of the *reparametrization trick* [1] $\mathbf{x}_n = \boldsymbol{\sigma}_n \odot \boldsymbol{\epsilon} + \boldsymbol{\mu}_n$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ to allow gradient calculation despite using sampling. For large datasets, using a single sample to approximate the expectation has shown to offer sufficient variability and greatly speeds up inference. The second term, acting as a regularizer on ϕ, is the Kullback-Leibler divergence between two Gaussians and can be given in closed differentiable form. Optimization can thus be performed by minibatch SGD. It has been shown that a complex decoder can lead to blurry reconstructions while a deep encoder can lead to uninformative latent variables [7, 8]. Treating the variance of the decoder distribution as a hyperparameter mitigates this issue and allows for an adjustable trade-off between the reconstruction loss and the regularizer. In this configuration, the model can be seen as an instance of the β-VAE as introduced in [9].

2.2. HMMVAE

To extend the modeling capabilities of the VAE towards datasets where every element is a time series, i. e. $\mathcal{Y}{=}\{\mathbf{Y}_1,\dots,\mathbf{Y}_N\}$, with $\mathbf{Y}_n{=}(\mathbf{y}_{n,1},\dots,\mathbf{y}_{n,T_n})$, $1{\leq}t{\leq}T_n$, in [3] it is proposed to incorporate a structured PGM prior for the latent code vector consisting of an HMM. The state at each time step is modeled by a sequence of discrete latent state variables $\mathbf{Z}{=}(z_1,\dots,z_T), 1{\leq}z_t{\leq}N_s\ \forall t$, where N_s is the number of states and the sequence index is omitted for brevity. Each state emission density is given by a Gaussian, while further parameters are inital state and transition probabilities:

$$p(\mathbf{x}_t|z_t{=}k) = \mathcal{N}(\mathbf{x}_t,\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k),$$
$$p(z_1;\boldsymbol{\pi}) = \prod_{i=1}^{N_s}\pi_i^{[z_1=i]},$$
$$p(z_t|z_{t-1};\mathbf{A}) = \prod_{i=1}^{N_s}\prod_{j=1}^{N_s} a_{ij}^{[z_{t-1}=i,z_t=j]},$$

where the Iverson bracket is used for state indexing. The full set of PGM parameters will be denoted by $\Omega{=}(\mathbf{A},\boldsymbol{\pi},\boldsymbol{\Sigma},\boldsymbol{\mu})$ with $\boldsymbol{\Sigma},\boldsymbol{\mu}$ denoting the Gaussian parameters for all classes.
Again, an approximate posterior has to be stated for inference, where a mean field approximation is used which still has to respect the Markov chain structure of the state sequence: $q(\mathbf{X},\mathbf{Z}){=}q(\mathbf{Z})\prod_{t=1}^{T} q(\mathbf{x}_n;\phi)$. All parameters of either the PGM and the NNs are assumed to be deterministic and training is again performed by a minibatch SGD of the ELBO, where the contribution of a single sequence is given by

$$\mathcal{L}^{(\text{HMMVAE})} = \mathbb{E}_{q(\mathbf{X},\mathbf{Z})}\left[\log\frac{p(\mathbf{Y},\mathbf{X},\mathbf{Z};\delta,\Omega)}{q(\mathbf{X},\mathbf{Z};\phi,\Omega)}\right].$$

The encoder $q(\mathbf{x}_t;\phi)$ and decoder $p(\mathbf{y}_t|\mathbf{x}_t;\delta)$ are modeled exactly as in eq. 2 and 1, respectively, due to intractability of closed form variational inference (VI). To calculate the ELBO, the state posterior $q(z_t)$ and joint posterior $q(z_{t-1},z_t)$ are needed. They can be calculated by extracting the optimal state sequence posterior from the ELBO:

$$\log(q^*(\mathbf{Z})) = \sum_{t=1}^{T}\sum_{j=1}^{N_s}[z_t{=}j]\mathbb{E}_{q(\mathbf{x}_t)}[\log\mathcal{N}(\mathbf{x}_t;\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j)]$$
$$+\sum_{t=2}^{T}\sum_{j=1}^{N_s}\sum_{i=1}^{N_s}[z_{t-1}{=}i,z_t{=}j]\log a_{ij} + \sum_{j=1}^{N_s}[z_1{=}j]\log\pi_i + \text{C}.$$

Since this term has the same structure as the log joint density of all states and emissions of a standard HMM, the needed posteriors can be calculated by the forward-backward algorithm. Making the replacements $q(z_t) \to [z_t{=}\hat{\jmath}_t]$ and $q(z_{t-1},z_t) \to [z_t{=}\hat{\jmath}_t, z_t{=}\hat{\imath}_{t-1}]$ with $\hat{\jmath}_t$ and $\hat{\imath}_{t-1}$ being Viterbi [10] estimates of the most probable current and previous state, respectively, is also possible. Finally, (semi-)supervised learning can be done as well by simply plugging in the true state labels instead. All necessary parameters (δ,ϕ,Ω) can now be trained using minibatch stochastic gradient ascent.

The AUD task can be tackled with this approach by stating a transition matrix structure where each hypothesized acoustic unit (AU) is implicitly represented by an internal three-state left-to-right topology while all AUs are connected. However, the number of AUs U has to be set as a hyperparameter, resulting in $N_s = 3U$ states in total.

2.3. Bayesian HMMVAE

The shortcomings of the HMMVAE are addressed here by embedding the PGM in a variational Bayesian framework. All PGM parameters Ω are now seen as random variables as well. All priors are chosen as conjugate w.r.t. the functional form of the conditional distributions containing the respective parameters. Most importantly, the AU active in each speech segment is now modelled as a categorical variable c_t and all labels of the n-th sequence make up the set $\mathbf{C}_n = (c_{n,1},\dots,c_{n,T_n})$. Assuming the DP as a prior for the AU distribution probability vector $\boldsymbol{\pi}^{(\text{DP})}$ allows the system to autonomously infer the number of AUs. In a DP, the number of acoustic units can in theory approach infinity. In practice however a maximum number of classes has to be given as a truncation parameter, i.e. $U < \infty$, which should be sufficiently large. For variational inference, the DP can either be approximated by a truncated stick-breaking prior as in [11, 4] or by a symmetric Dirichlet distribution. According to [12], the latter should be preferred and our conducted experiments lead to the same conclusion. Thus, the model for the acoustic units becomes

$$p\left(c,\boldsymbol{\pi}^{(\text{DP})}\right) = \left(\prod_{k=1}^{U}(\pi_k^{(\text{DP})})^{[c=k]}\right)\cdot\text{Dir}\left(\boldsymbol{\pi}^{(\text{DP})};\frac{\alpha}{T}\mathbf{1}_U\right),$$

with $\mathbf{1}_U$ being the U-dimensional vector consisting of only ones.

The unit sequence strongly influences the state sequence Z. Furthermore, the transitions between the unit-HMMs have to be modelled as in [4].

Similar to the HMMVAE, the latent emission distribution is a Gaussian, but with either a Normal-Wishart prior for full covariance matrices or a Normal-Gamma prior for the diagonal case placed on its parameters. With this structure, the approximate posterior distribution $q(\Omega;\lambda)$ of the global PGM parameters shows a conjugacy structure which depends on a set λ of variational parameters. Assuming a mean field approximation

$q(\mathbf{X}, \mathbf{Z}, \mathbf{C}, \Omega; \phi, \lambda) = q(\mathbf{Z}, \mathbf{C}) \prod_{t=1}^{T} q(\mathbf{x}_n; \phi) q(\Omega)$ similar to the HMMVAE case, the lower bound training objective for a single sequence can be split into three different parts, where λ_0 are the parameters of the PGM prior distributions and $\mathrm{H}(\cdot)$ denotes the entropy of a given distribution:

$$\mathcal{L}^{\text{(BHMMVAE)}} = \mathbb{E}_{q(\mathbf{X},\mathbf{Z},\mathbf{C},\Omega)} \left[\log \frac{p(\mathbf{Y}, \mathbf{X}, \mathbf{Z}, \mathbf{C}, \Omega; \delta, \lambda_0)}{q(\mathbf{X}, \mathbf{Z}, \mathbf{C}, \Omega; \phi, \lambda)} \right]$$
$$= \mathbb{E}_{q(\mathbf{X};\phi)} \left[\log p(\mathbf{Y}|\mathbf{X}; \delta) \right] + \mathrm{H}(q(\mathbf{X}; \phi))$$
$$+ \mathbb{E}_{q(\mathbf{X};\phi)} \left[\mathbb{E}_{q(\mathbf{Z},\mathbf{C},\Omega;\lambda)} \left[\log \frac{p(\mathbf{X}, \mathbf{Z}, \mathbf{C}, \Omega; \lambda_0)}{q(\mathbf{Z}, \mathbf{C}, \Omega; \lambda)} \right] \right]. \qquad (3)$$

The inner expectation value of the third term is the ELBO for a Bayesian HMM as given in [4] and only the emission probabilities contain $\mathbf{X}$. Thus, the update for each minibatch can be separated into a sequence of steps:

1. Code posterior samples $\mathbf{x}_n^l \sim q(\mathbf{x}_n) \forall n$ are drawn. As before, one sample per time index is usually sufficient.
2. The latent PGM variables are inferred with the forward-backward (FB) or Viterbi algorithm, and the latent parameters are trained with a VI method using only the third term.
3. The decoder network parameters δ are learned using only the first term and the samples.
4. The encoder network parameters are trained using all three terms, but only the emission model part is needed from the third term.

Due to the latent PGM parameters affecting every sequence, standard VI would lead to a batch variational EM scheme, where their variational parameters would only be updated after visiting all sequences of the whole dataset. Since the NN parameters are still trained by minibatch SGD, this would lead to a mismatch, where the NN training would find vastly different conditions after each batch update, leading to slow convergence.

In order to increase the matching between the two training algorithms, SVI [6] is used for optimizing the PGM parameters, resulting in two different gradient-based learning algorithms which visit each minibatch simultaneously. The core idea behind SVI is that the ordinary gradient of the ELBO w.r.t. the variational parameters would involve calculating their Fisher information matrix $\mathrm{I}(\lambda)$, which is computationally expensive. However, it has been shown in [13] that taking the so called natural gradient $\tilde{\nabla}_\lambda \mathcal{L} = \mathrm{I}(\lambda)^{-1} \nabla_\lambda \mathcal{L}$ works better anyway since it allows optimization in the space of distributions instead of the parameter space. When the distributions are parametrized as instances of exponential families, this calculation exactly cancels out the troublesome matrix, leaving the expression

$$\tilde{\nabla}_\lambda \mathcal{L} = \hat{\lambda}_n - \lambda,$$

where $\hat{\lambda}_n$ is the optimal estimate for the variational parameters for the current example. Plugging this result into the gradient ascent rule with τ as the learning rate leads to the remarkably simple result

$$\lambda_{n+1} = \lambda_n + \tau \left(\hat{\lambda}_n - \lambda_n \right) = (1 - \tau) \lambda_n + \tau \hat{\lambda}_n$$

i.e. the update rule is a first order lowpass filter between the last value and the best estimate for the current example. Extending this result to a minibatch algorithm, one arrives at

$$\hat{\lambda}_m = \frac{N}{M_m} \sum_{n \in \mathcal{M}_m} \hat{\lambda}_n,$$

i.e. the arithmetic mean over all examples in the current minibatch with index m, the index set $\mathcal{M}_m$ and $M_m = |\mathcal{M}_m|$ as the number of examples in the current minibatch, multiplied by N, the total number of examples, to arrive at a bias-free estimate [6].

3. Experiments

3.1. Setup

The proposed model is evaluated through the task of unsupervisedly learning acoustic units on the TIMIT database [14] and on the NCHLT Xitsonga corpus [15] as used in the 2015 Zero Resource Challenge [16] and compared to the HMMVAE and the Bayesian Gaussian mixture model (GMM)-HMM as given in [4]. For the latter, the authors' reference implementation [17, (AMDTK)] is used.

In the case of TIMIT, training and testing is carried out on the complete datasets including the dialect sentences (SA), thus using all 6300 utterances, with 100 randomly chosen utterances used for cross validation. Likewise, the whole Xitsonga database is used, containing 2 hours and 30 minutes of speech in total.

For both databases, each utterance is transformed into the log-mel domain using an STFT window size of $25\,\mathrm{ms}$ with a window overlap of $10\,\mathrm{ms}$, and a filterbank consisting of 40 mel filters. Furthermore, delta and delta-delta features are calculated, totaling in a feature dimension of 120, and mean-variance normalization is applied.

For evaluation, the normalized mutual information (NMI) as described in [4] is used: After training, each frame has an AU label assigned by the AUD system and a phone label given by the ground-truth transcription, respectively. By means of counting, a normalized confusion matrix can be calculated as an estimate for the joint probability between AUs and phones and the NMI is defined as $\mathrm{NMI} = \mathrm{I}(U; P)/\mathrm{H}(P)$, where the phones are P, the AUs are U, the numerator is the mutual information between them and the denominator is the entropy of the phones, resulting in a measure of their statistical dependency normalized to the interval $[0, 1]$. Clearly, the higher the NMI, the better. Further note that the NMI depends on the number of AUs. Thus, only NMI values achieved with roughly the same number of AUs are comparable.

Additionally, to incorporate another more intuitive measure, the *equivalent* phone error rate (PER) is calculated. Firstly, each AU is mapped to the phone with which it overlaps most according to the confusion matrix and uninterrupted sequences of the same unit are contracted to only one occurrence. Then, the PER is calculated as $\mathrm{PER} = (\mathrm{Sub.} + \mathrm{Ins.} + \mathrm{Del.})/\mathrm{Tot.})$, i.e. the edit distance between the mapped AU transcription and the ground truth transcription, normalized to the total number of phones instances in the ground truth transcription. This measure is an adaptation to the many-to-one word error rate (WER) given in [18]. Both measures are already used in [3].

The model architecture is as follows: Each AU to be found is modeled as a three-state HMM and an overall transition matrix is constructed by using the AU weights as given by the DP as the transition probabilities between different units. Both NNs show a feed-forward structure with two hidden layers of 512 units each, while the dimensionality of the latent code is chosen as $D_x = 32$. The latent GMM prior parameters are $\mathbf{m}_0 = \mathbf{0}, \kappa_0 = 1, \nu = D_x{+}1, \mathbf{W}_0 = \mathbf{I}$ in case of a Normal-Wishart prior for full covariance matrices. In the case of di-

Table 1: *Comparison of AUD results on the TIMIT database*

model/init	cov	τ	ω_0	PER	NMI	#AU
GMM-HMM	D	-	-	65.42	37.84	72
HMM-VAE	F	-	-	58.54	43.90	72
BHMMVAE pre-train	D	0.0010	1.000	58.74	45.08	72
	D	0.0010	0.100	**56.57**	**45.97**	85
	D	0.0010	0.010	57.31	44.58	87
	D	0.0100	0.010	63.12	38.81	37
	F	0.0010	0.010	62.15	40.13	47
	F	0.0005	0.010	60.91	42.53	47
	F	0.0001	0.010	62.15	40.13	47
BHMMVAE cluster	F	0.005	1.000	58.81	45.49	89
	F	0.005	0.010	57.91	**45.92**	90
	F	0.010	1.000	58.26	44.71	87
	F	0.005	0.001	**57.41**	45.05	89

Table 2: *Comparison of AUD results on the Xitsonga database*

model	cov	τ	ω_0	PER	NMI	#AU
GMM-HMM	D	-	-	72.60	35.00	69
HMM-VAE	F	-	-	61.90	37.60	69
BHMMVAE pre-train	D	0.001	1.000	62.65	37.08	69
	F	0.001	1.000	62.64	37.08	69
	D	0.001	0.100	62.09	**40.06**	100
	D	0.001	0.010	62.69	39.85	96
	D	0.001	0.001	62.47	39.14	98
	D	0.005	0.010	64.45	37.72	56
	F	0.001	0.010	62.57	37.06	100
	F	0.005	0.010	**61.97**	39.67	**61**

agonal covariance matrices, a Normal-Gamma prior is used, where $\mathbf{m}_0 = \mathbf{0}, \kappa_0 = 1, \alpha_0 = 1, \boldsymbol{\beta}_0 = 1$. In both cases, the parametrization is done as in [19]. Since the truncated stick-breaking approximation of the DP has shown to be unstable in the experiments, a symmetric Dirichlet distribiution approximation as in [12] with a truncation of $U{=}100$ is used instead and the concentration parameter ω_0 is varied in the experiments. Furthermore, two different initialization strategies for the model are compared:

- The **pre-train** strategy is taken from [3]. Here, a randomly generated AU alignment with a unit duration of 6 frames and a state duration of 2 frames is generated for each utterance. Then, pseudo-supervised training is performed for 20 epochs with the random alignment as a tentative target label sequence.
- The **cluster** strategy consists of an up-front training of a standard VAE for 20 epochs to initialize the encoder and decoder, subsequently encoding the feature vectors of each utterance into the code space and performing a k-means clustering using $3 \cdot 100$ clusters. Then, a random assignment between clusters and states is performed with the cluster mean and assignment count used as the posterior mean and posterior count variables of the corresponding state, respectively.

The model training is performed with a constant minibatch size of $M_m{=}16$ utterances and terminated after 100 epochs. *Adam* [20] with a fixed step size of 0.001 and gradient clipping is used for training the NN parameters, while the PGM parameters are simultaneously updated in each minibatch using SVI. Since these two learning processes are running in an interleaved manner, their convergence behaviour should be matched for good results. We therefore investigate the impact of the SVI learning rate τ.

3.2. Results

The results for TIMIT are given in Table 1. The covariance matrix type is either diagonal (D) or full (F). In addition to the measures explained above, the resulting number of units is given for each result. In the first two results rows the performance of the GMM-HMM and HMMVAE are given as a reference [3], where the latter had been initialized with the former. The Bayesian hidden Markov model variational autoencoder (BHMMVAE) is a stand-alone system, which does not require the GMM-HMM for initialization. In the third results row $U{=}72$ and $\omega_0{=}1.0$ were fixed to achieve the same number of units as the GMM-HMM and HMMVAE for ease of comparison. The NMI is improved over those, but other combinations of parameters, especially with $\omega_0{<}1$, lead to even better results. Moreover, the model is fairly robust w.r.t. variations in the concentration parameter. With the pre-train strategy, a high value of the SVI learning rate τ leads to results where more units are discarded and the performance suffers. Note that full covariance matrices do not work well with the pre-train strategy. This is most likely caused by the increased number of parameters, which results in a model too flexible. A successful training of full covariance matrices is possible when increased guidance is provided by the cluster initialization strategy. The results come close to the better outcomes of the diagonal matrices in combination with the pre-train strategy, but are unable to improve despite the higher computational complexity.

On the Xitsonga setup, all experiment runs for the BHMMVAE are therefore performed using the pre-train strategy. In the same manner as done for TIMIT, $U{=}72$ and $\omega_0{=}1$ are set for the first two rows to achieve full comparability. Overall, BHMMVAE achieves roughly the same performance as the HMMVAE or outperforms it w.r.t. the NMI, while the PER is usually a bit higher. Taking both databases into account, the best choice seems to be the pre-train strategy in combination with diagonal covariance matrices, a small learning rate and a moderate to low concentration parameter, i. e. $\tau = 0.001$ and $0.1 \leq \omega_0 \leq 0.01$.

4. Conclusions

We have developed a full Bayesian treatment of the HMM-VAE, where the training relies on stochastic variational inference using the natural gradient. We have carried out experiments on AUD on two databases, TIMIT and Xitsonga. The main findings are: (a) the algorithm is able to find automatically a reasonable number of AUs and can be trained stand-alone from scratch without guidance by a GMM-HMM, and, (b) NMI and PER are comparable to those obtained by the HMMVAE or even better.

5. Acknowledgement

The work reported here was supported by Deutsche Forschungsgemeinschaft (DFG) under contract no. Ha 3455/12-1 within the priority program 1527 Autonomous Learning, and contract no. Ha 3455/15-1.

We would like to thank Lucas Ondel and Lukáš Burget of Brno University of Technology for valuable discussions.

6. References

[1] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[2] 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*, 2016, pp. 2946–2954.

[3] J. Ebbers, J. Heymann, L. Drude, T. Glarner, R. Haeb-Umbach, and B. Raj, "Hidden markov model variational autoencoder for acoustic unit discovery," in *INTERSPEECH 2017, Stockholm, Schweden*, August 2017.

[4] L. Ondel, L. Burget, and J. Cernocky, "Variational inference for acoustic unit discovery," *Procedia Computer Science*, vol. 81, pp. 80–86, 2016.

[5] Y. W. Teh, "Dirichlet processes," in *Encyclopedia of Machine Learning*. Springer, 2010.

[6] M. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic Variational Inference," *ArXiv e-prints*, Jun.

[7] H. Zheng, J. Yao, Y. Zhang, and I. W. Tsang, "Degeneration in VAE: in the Light of Fisher Information Loss," *ArXiv e-prints*, Feb. 2018.

[8] A. A. Alemi, B. Poole, I. Fischer, J. V. Dillon, R. A. Saurous, and K. Murphy, "Fixing a Broken ELBO," *ArXiv e-prints*, Nov. 2017.

[9] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," 2016.

[10] G. D. Forney, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[11] D. M. Blei and M. I. Jordan, "Variational inference for Dirichlet process mixtures," *Bayesian Anal.*, vol. 1, no. 1, pp. 121–143, 03 2006.

[12] K. Kurihara, M. Welling, and Y. W. Teh, "Collapsed variational Dirichlet process mixture models," in *Proceedings of the International Joint Conference on Artificial Intelligence*, vol. 20, 2007.

[13] S.-I. Amari, "Natural gradient works efficiently in learning," *Neural Comput.*, vol. 10, no. 2, pp. 251–276, Feb. 1998.

[14] 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.

[15] N. J. de Vries, M. H. Davel, J. Badenhorst, W. D. Basson, F. de Wet, E. Barnard, and A. de Waal, "A smartphone-based ASR data collection tool for under-resourced languages," *Speech Communication*, vol. 56, pp. 119 – 131, 2014.

[16] M. Versteegh, R. Thiolliere, T. Schatz, X.-N. Cao, X. Anguera, A. Jansen, and E. Dupoux, "The zero resource speech challenge 2015." in *Interspeech*, 2015, pp. 3169–3173.

[17] "AMDTK," http://amdtk.readthedocs.io, accessed: 2017-07-11.

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

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

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.