*INTERSPEECH 2016*
September 8–12, 2016, San Francisco, USA

# Unsupervised Learning of Acoustic Units Using Autoencoders and Kohonen Nets

*Vikramjit Mitra, Dimitra Vergyri, Horacio Franco*

Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA
{vikramjit.mitra, dimitra.vergyri, horacio.franco}@sri.com

## Abstract

Often, prior knowledge of subword units is unavailable for low-resource languages. Instead, a global subword unit description, such as a universal phone set, is typically used in such scenarios. One major bottleneck for existing speech-processing systems is their reliance on transcriptions. Unfortunately, the preponderance of data becoming available everyday is only worsening the problem, as properly transcribing, and hence making this data useful for training speech-processing models, is impossible. This work investigates learning acoustic units in an unsupervised manner from real-world speech data by using a cascade of an autoencoder and a Kohonen net. For this purpose, a deep autoencoder with a bottleneck layer at the center was trained with multiple languages. Once trained, the bottleneck-layer output was used to train a Kohonen net, such that state-level ids can be assigned to the bottleneck outputs. To ascertain how consistent such state-level ids are with respect to the acoustic units, phone-alignment information was used for a part of the data to qualify if indeed a functional relationship existed between the phone ids and the Kohonen state ids and, if yes, whether such relationship can be generalized to data that are not transcribed.

**Index Terms**: unsupervised learning, Kohonen nets, speech recognition, low-resource languages, acoustic unit discovery.

## 1. Introduction

Creating speech technologies across multiple languages is often difficult and labor intensive. Before creating speech-processing tools for a language, several requirements may have to be met. Firstly, the language having a written form, which enables orthographic transcription, facilitates speech-processing system creation. Secondly, having some transcribed data is immensely useful, as most speech-processing tools are based on supervised learning, with clear input-output pairs needed to train and deploy a reasonable model.

In a language, words are usually represented in terms of phonemes, which are the basic phonological unit of a language, and phoneme inventories vary from language to language [1]. With social networking, affordable high-speed internet, and open sharing of multimedia content, the past decade has witnessed increased interest in speech tools geared toward multi-language processing capabilities. Basic speech tools exist for the world's popular languages; but for low-resource and less frequently used languages, building speech-processing systems is quite challenging. Availability of transcribed material is often limited, as it necessitates availability of language experts who know the target language well, and often such resources are unavailable.

Audio materials are usually more readily available, hence the speech tools that leverage audio-only material have the edge in dealing with new languages. Several studies have investigated ways to directly learn subword units in an unsupervised manner from the speech signal. Studies in [2, 3] proposed unsupervised acoustic modeling through segmentation, clustering, and modeling each cluster, where *a priori* knowledge about the number of subword unit to be learned was assumed to be known. In [4], a single-state hidden Markov model (HMM) was trained by using the entirety of the available acoustic data, and then iterative state-splitting was performed based on an objective function. Pattern discovery was used in [5], which trained HMMs for each found pattern in the acoustic data. In [6], an unsupervised model was proposed that simultaneously segmented the speech signal, discovered subword units and learned an HMM for each induced acoustic unit. In [7], the authors assumed that the training data had been word transcribed and that some relationship existed between the orthography and pronunciation of the language. To discover acoustic units from context-dependent grapheme models, [7] used spectral clustering that worked on full HMM models. In [8], an autoencoder (AE)-based unsupervised acoustic-unit discovery was proposed, in which the authors showed that an AE representation is better than Gaussian posteriograms in a spoken-query classification task. In that work, the AE decisions were discretized through thresholding.

In this work, we investigate building bottleneck-deep autoencoder (BN-DAE) networks that learn an acoustic space in an unsupervised, data-driven manner. Speech from multiple languages is used to train the BN-DAE. The target language on which results will be evaluated is not included in the training languages. Once the BN-DAE network is learned, the bottleneck (BN) features are used to train Kohonen nets. Different Kohonen nets (KN) are trained, with the networks having different numbers of target neurons and different BN feature time-contextualization. Kohonen nets [7] employ competitive learning, and are a form of a self-organizing map that is trained in an unsupervised, data-driven manner. The Kohonen net's role is assigning state-level ids to the BN features such that a discrete representation is created from the continuous BN feature space. Once the KNs are learned, they are used to decode speech signals and produce a sequence of hypothesized state ids. We use a small amount of data with phone alignments to obtain the conditional distribution of a phone given a KN state id. This conditional distribution is used to predict phone ids given KN ids. The results are reported in terms of frame-level phone accuracy, which indicates that the proposed approach can learn states similar to phone units in an unsupervised manner.

The paper is structured as follows. First, in Section 2, we briefly describe the dataset used in our experiments. In Section

http://dx.doi.org/10.21437/Interspeech.2016-1374

3, we present the BN-DAE system and how it was trained. In Section 4, we present the KNs used in this work and briefly describe how they were trained. In Section 5, we show the results from our experiments. Finally, in Section 6, we present our conclusions.

## 2. Dataset and acoustic features

The speech data used to train the BN-DAE and KN models was taken from seven language training sets, from various sources: Assamese (BABEL); Bengali (BABEL); Dari (Transtac); Egyptian Arabic (Callhome); English (Fisher); Mandarin (GALE); and Spanish (Callhome). We used the following Babel data releases: Amharic, IARPA-babel307b-v1.0b; Assamese, IARPA-babel102b-v0.5a; Bengali, IARPA-babel103b-v0.4b; and Pashto, IARPA-babel104b-v0.4bY. FullLP training sets were used. In total, this comprised approximately 650 hours of audio data in the seven languages. All data was sampled at 8 kHz. Note that neither speaker- nor language-level information was ever used in any of the processing outlined in this work. The raw audio data was parameterized as gammatone filterbank energy (GFB) acoustic features. Gammatone filters are a linear approximation of the auditory filtering performed in the human ear. The GFBs were extracted by using SRI International's implementation of a time-domain gammatone filterbank, which contained 40 channels that were equally spaced on the equivalent rectangular bandwidth (ERB) scale, between 150 Hz and 3750 Hz. For the acoustic features, the analysis window was 25.6 ms, with a frame rate of 10 ms. The GFBs used 15th power root nonlinear compression.

The performance of the proposed approach was evaluated on an unseen language: Amharic, which was taken from the BABEL program. Note that the Amharic phone sets were mapped to a broad phone set, such that all the Amharic phones were a subset of the phones in the seven-language training set. We had approximately seven hours of data for Amharic, which were split three ways: one hour for learning the conditional distribution of a phone given a KN id, one hour as development data, and the remaining five hours for testing.

## 3. Bottleneck-deep autoencoder (BN-DAE) system

The BN-DAE system was a five-hidden-layer, fully connected DNN system, with the third hidden layer containing a bottleneck of eighty neurons. The remaining hidden layers had 1024 neurons. The hidden layers had sigmoid activations, whereas the output layer had linear activation. The BN-DAE was trained by using mean squared error (MSE) backpropagation. The input to the BN-DAE system was 40 GFBs with a splicing of 11, resulting in 440 dimensional features. The output was the same 40 GFBs, but with a splicing of three. A block diagram of the BN-DAE and KN system is shown in Figure 1.

The BN-DAE system was trained using the mean squared error criteria, with Gaussian random Bernoulli (GRBM) pre-training. The networks were trained by using an initial few iterations with a constant learning rate of 0.09, followed by learning rate decrease by a factor of 0.8 based on cross-validation error decrease. Training stopped when no further significant reduction in cross-validation error was noted or when cross-validation error started to increase. Backpropagation was performed by using stochastic gradient descent with a mini-batch size of 512. The BN features from the BN-DAE were then used as the input to the KN.

## 4. Kohonen nets (KNs)

A Kohonen net is a type of artificial neural network that uses unsupervised learning to generate a low-dimensional discretized abstraction of relatively high dimensional input observations. They are also popularly known as self-organizing maps (SOMs), which employ competitive learning that preserves the topological properties of the input feature space [10].

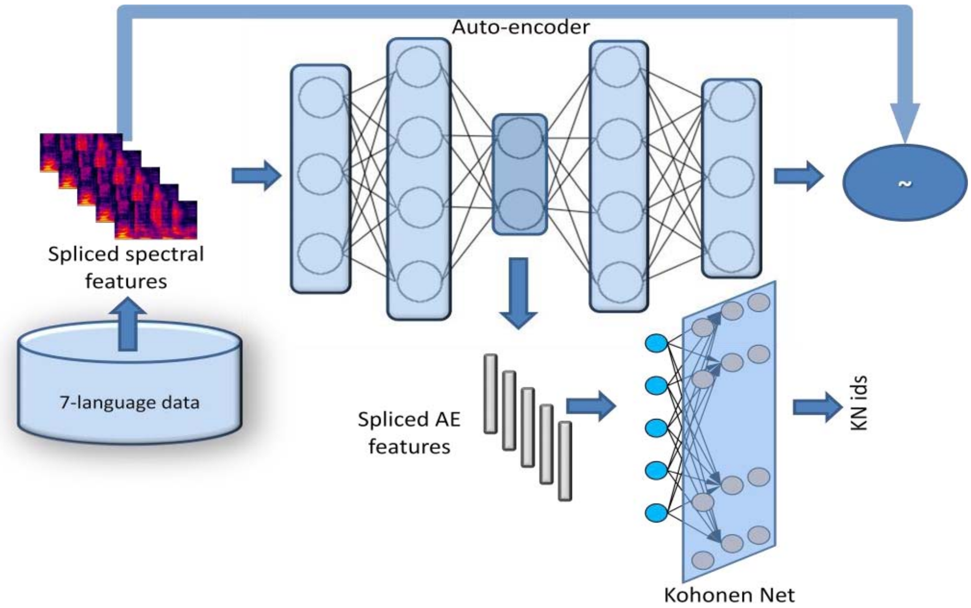

Figure 1: *The unsupervised acoustic unit discovery system using a bottleneck-deep autoencoder (BN-DAE) system and Kohonen nets.*

Like other neural nets, KNs consists of neurons that are parameterized by their weights and biases. The goal of the KN is to differentiate excitation signals, and each neuron learns to respond similarly for similar input excitation signals. Such topological partitioning of the input space is learned through a competitive learning approach, which allows only one neuron to be active given an input excitation signal, hence inhibiting the excitation of the other neurons.

In this work, the weights of the neurons were initialized with small random numbers, and then the training was performed in mini-batches. Each input sample was presented to the KN multiple times through random alteration of the mini-batches. For each training example, the network computed its Euclidean distance to all weight vectors. The neuron whose weight vector was most similar won the competition. The weights of the winning neuron and its neighbors in the SOM lattice were adjusted toward the input vector. The weights of the winning neuron were adjusted by using the Kohonen learning rule, which is stated below for the case where the $i^{th}$ neuron wins:

$$W_i(q) = W_i(q-1) + \psi(i,j,q)\boldsymbol{\alpha}(q)\{p(t) - W_i(q-1)\} \quad (1)$$

Where $q$ is the step index; $j$ is a neighboring neuron; $p(t)$ is the input feature to the KN; and $\psi$ is the neighborhood function that specifies the distance between neighboring neurons $i$ and $j$ in step $q$. Note that the neighborhood function $\psi$ shrinks with time, where at the onset a broad neighborhood is considered, and with training, the neighborhood map is reduced to only the immediately neighboring neurons.

The goal of the KN training was to ensure that the winning neuron was more likely to win the competition the next time a similar vector was presented to the network, and less likely to win when a very different input vector was presented. As more and more inputs were presented, each neuron in the layer closest to a group of input vectors adjusted its weight vector toward those input vectors. Eventually, with sufficient number of neurons, the network was able to learn clusters, where every cluster of similar input observations would have a winning neuron when a vector belonging to that cluster was presented. At the end of training, the competitive network learned to categorize the input vectors it saw, with the outputs being cluster ids.

## 5. Experiments and results

The BN-DAE networks were trained with 40-dimensional GFB features by using the architecture depicted in Figure 1. Note that the data used for training the BN-DAE system did not contain any target language data (in this case, Amharic). The bottleneck layer had eighty neurons. We explored the reliability of the bottleneck features by training a DNN acoustic model, where a five-hidden-layer acoustic model was trained with the seven languages (Assamese, Bengali, Dari, Egyptian Arabic, English, Mandarin, and Spanish) and tested with Amharic. Compared to a baseline DNN model trained with GFB features, the BN-DAE features were found to reduce phone error rates, indicating that the bottleneck layers of the BN-DAE network could learn meaningful information about the acoustic data for speech recognition tasks. We also trained a stacked bottleneck (SBN) system similar to that in [11], and the results were similar as for the BN-DAE system.

The BN features extracted from the BN-DAE model were then time contextualized (spliced) and then used to train the KN model. We investigated KNs with sixty and eighty classes, where the features were spliced over 15 frames with and without frame-skipping. In the latter case with frame skipping (where every other frame was skipped), the dimension of the input BN-DAE features were almost reduced to half. The KNs were trained with a mini-batch size of 1000, with 200 epochs per mini-batch and two-fold training over the mini-batches. The Kohonen learning rate was selected as 0.01.

Once the KNs were trained, the BN features extracted from the Amharic data were used to decode the KN models, which resulted in a sequence of KN ids. In order to map the learned KN ids to the Amharic phone sets, we used an hour of Amharic data with transcriptions to generate forced alignments. The KN ids and phone alignments were then used to generate a K2P lookup table of conditional probability distribution $p(a|k)$, where $a$ is a phone label, and $k$ is the KN id. Note that we observed altogether 33 unique phone tokens in Amharic, with three tokens corresponding to non-speech units SIL, SPN, and LAU.

The KN outputs were treated as an independent process, where for each frame $n$, a phone id was assigned by employing the K2P lookup table using:

$$\hat{a}_n = argmax_i p(a|k_n) \quad (2)$$

Note that for the decision in (2), neither the language model nor the temporal dependence of the KN ids with the phone tokens were used. The process is outlined in Figure 2.

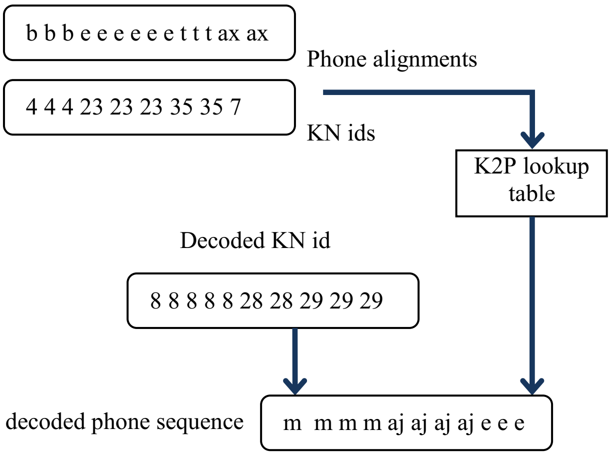

Figure 2: *Schematics of K2P lookup table creation and KN-id decoding.*

The performance of the system is evaluated with respect to frame accuracy, which gives the proportion of the frames that were recognized correctly, compared to the alignments of the Amharic test as references. Table 1 presents the frame-recognition accuracy obtained from the development (one hour) and test (five hours) sets from the four KN models trained in our experiments.

Table 1. *Frame-recognition accuracy from different KN systems.*

| | #class | splice | skip | dev | test |
|---|---|---|---|---|---|
| KN-1 | 60 | 7 | 1 | 55.8 | 57.3 |
| KN-2 | 60 | 7 | 2 | 56.0 | 57.5 |
| KN-3 | 80 | 7 | 1 | 56.5 | 58.0 |
| KN-4 | 80 | 7 | 2 | 56.1 | 57.7 |

To assess the performance of the above system, we also decoded the Amharic test data by using a DNN phone-recognition system trained with the seven languages using the BN-DAE features. The DNN model had five hidden layers with 1200 neurons. The model was trained with crossword triphones, where altogether 4945 context-dependent (CD) states were used. Table 2 shows the frame-recognition accuracy from the DNN phone-recognition model, that uses a bigram language model. Note that the phone recognition model was trained with the seve-language training set, where Amharic data was not used for training. In Table 2, we also show the result from an updated KN-3 model (the best model in Table 1), which was rescored using K2P lookup table that was learned, avoiding 30 ms of information from the phone boundaries.

Table 2. *Frame-recognition accuracy from the best KN system and a DNN based phone recognition model trained with BN-DAE features*

| | splice | skip | dev | test |
|---|---|---|---|---|
| KN-3 updated K2P lookup | 7 | 1 | 68.3 | 70.6 |
| DNN-phone model | 7 | 1 | 74.6 | 76.1 |

Table 2 shows that the KN-3 model with an updated lookup table obtains Amharic phone recognition quite competitively with respect to the DNN phone-recognition model. Note that in this approach, no language model (i.e., phone-sequence probabilistic relations between phone units) was used, nor has KN-id sequence-level information been used. Such information when used can improve phone-detection accuracy.

## 6. Conclusions

In this work, we proposed an approach to learn acoustic units in an unsupervised fashion from the speech signal. Firstly, an acoustic subspace is learned through a bottleneck-deep autoencoder (BN-DAE) model that separates the acoustic units and hence simplifies the task of KN-based acoustic-unit discovery. The BN-DAE system had eighty neurons in its bottleneck layer, but neither the number of neurons in the BN layer nor the number of hidden layers in the BN-DAE system was optimized. GFB features were used to train the BN-DAE system.

In the future, we plan to investigate other robust features such as modulation features [12, 13], damped oscillator features [14], etc., which have demonstrated better or competitive performance with respect to GFBs and hence could either provide better BN-DAE systems or be used to learn BN-DAE systems that capture complementary information that could improve performance when combined.

We also plan to investigate KN models with larger numbers of target classes trained with BN-DAE systems learned from different acoustic features. We further plan to investigate temporal relationships across learned KN ids and to leverage phone-based language models to improve phone-detection performance. We also plan to investigate KN-id-based keyword detection, in which case the KN-id sequences could be used directly to search for keywords.

## 7. Acknowledgements

This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via the Air Force Research Laboratory. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, AFRL, or the U.S. Government.

This effort uses the IARPA Babel Program Amharic language collection release IARPA-babel307b-v1.0b. We used the following Babel data releases: Amharic, IARPA-babel307b-v1.0b; Assamese, IARPA-babel102b-v0.5a; Bengali, IARPA-babel103b-v0.4b; and Pashto, IARPA-babel104b-v0.4bY. FullLP training sets were used.

## 8. References

[1] C-Y. Lee, T. J. O'Donnell and and J. Glass, "Unsupervised lexicon discovery from acoustic input," *Transactions of the Association for Computational Linguistics*, Vol. 3, pp. 389–403, 2015.

[2] C-H. Lee, F. Soong, and B-H. Juang, "A segment model based approach to speech recognition," in *Proc. of ICASSP*, pp. 501–504, 1988.

[3] A. Garcia and H. Gish, "Keyword spotting of arbitrary words using minimal speech resources," in *Proc. of ICASSP*, pp. 949–952, 2006.

[4] B. Varadarajan, S. Khudanpur and E. Dupoux, "Unsupervised learning of acoustic sub-word units," in *Proc. of ACL-08: HLT, Short Papers*, pp. 165–168, 2008.

[5] A. Jansen and K. Church, "Towards unsupervised training of speaker independent acoustic models," in *Proc. of Interspeech*, pp. 1693–1696, 2011.

[6] C-Y. Lee and J. Glass, "A nonparametric Bayesian approach to acoustic model discovery," in *Proc. of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers*, Vol. 1, 2012.

[7] W. Hartmann, A. Roy, L. Lamel and J-L. Gauvain, "Acoustic unit discovery and pronunciation generation from a gapheme-based lexicon," in *Proc. of ASRU*, 2013.

[8] L. Badino, C. Canevari, L. Fadiga and G. Metta, "An auto-encoder based approach to unsupervised learning of subword units," in *Proc. of ICASSP*, 2014.

[9] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybernetics* 43 (1), pp.59–69, 198). doi:10.1007/bf00337288.

[10] T. Kohonen, *Self-Organizing Maps. Series in Information Sciences*, $2^{nd}$ *Ed.* vol. 30, Heidelberg: Springer, 1997.

[11] M. Karafiát, F. Grézl, L. Burget, I. Szöke and J. Cernocký "Three ways to adapt a CTS recognizer to unseen reverberated speech in BUT system for the ASpIRE challenge," *Proc. of Interspeech*, pp. 2454–2458, 2015.

[12] V. Mitra, H. Franco, M. Graciarena and A. Mandal, "Normalized amplitude modulation features for large vocabulary noise-robust speech recognition," *Proc. of ICASSP*, pp. 4117–4120, Japan, 2012.

[13] V. Mitra, H. Franco, M. Graciarena and D. Vergyri, "Medium duration modulation cepstral feature for robust speech recognition," in *Proc. of ICASSP*, pp. 1768–1772, Florence, 2014.

[14] V. Mitra, H. Franco and M. Graciarena, "Damped oscillator cepstral coefficients for robust speech recognition," in *Proc. of Interspeech*, pp. 886–890, Lyon, 2013.