IDIAP RESEARCH REPORT

SPEECH MODELING USING SPARSE AUTOENCODERS

Selen Hande Kabil Hervé Bourlard

Idiap-RR-11-2022

AUGUST 2022

Centre du Parc, Centre du Parc, Rue Marconi 19, CH - 1920 Martigny
T +41 27 721 77 11 F +41 27 721 77 12 info@idiap.ch www.idiap.ch

SPEECH MODELING USING SPARSE AUTOENCODERS

Selen Hande Kabil[1,2] *and Hervé Bourlard*[1,2]

[1]Idiap Research Institute, Martigny, Switzerland
[2]École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

ABSTRACT

Dictionary learning and shallow overcomplete autoencoders with ℓ_1-norm penalty imposed on the hidden unit activations exhibit similar mathematical properties, sharing the same goal of projecting data in high dimensional spaces. Exploiting this connection and starting from a state-of-the-art lattice-free MMI (LF-MMI) speech recognition system on the AMI corpus, we investigate different ways to further improve this baseline system by projecting the acoustic model output features in higher dimensional sparse representation spaces. While the resulting recognition system seems to leverage at the baseline performance, we observe that higher dimensional sparse representation space is able to catch correlations among senone units. Detailed analysis and discussions are provided in the paper.

Index Terms— Speech modeling, dictionary learning, sparse autoencoder, representation learning

1. INTRODUCTION

As a consequence of the parsimonious hierarchical nature, class-specific low-dimensional factors exist in speech features [1, 2]. To model a phenomenon with low-dimensional causative factors, two methods are stated in [3]: (1) low-rank modeling to obtain individual compressed representations for each factor's subspace, (2) sparse modeling to obtain common high-dimensional sparse representation where all factors lie together in different subspaces.

In [4], the impact of low-rank and sparse modeling on acoustic model outputs (e.g., posterior features) for the speech recognition task is investigated. In low-rank modeling (Fig. 1), posterior features are projected on the manifolds of the underlying senone classes using class-specific undercomplete dictionaries. Each senone has its own undercomplete dictionary, which is learned on the posterior features belonging to that senone class. This requires senone alignments and grouping posteriors based on the correct senone class information. In sparse modeling (Fig. 1), posterior features are projected on an overcomplete dictionary, which is formed by concatenating all class-specific undercomplete dictionaries together. In [4], the projected posteriors are reported to be more accurate targets for learning better acoustic models, as they only keep the information about the correct underlying senone. However, this approach requires prior knowledge about the senone class information. Thus, it is not easily scalable for large vocabulary speech recognition tasks.

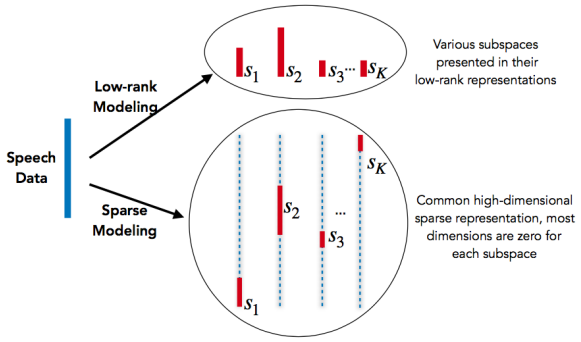


Fig. 1: Sparse and low-rank modeling of speech data (acoustic model output frame) with different senone classes, s stands for senone class. Figure adapted from [4].

In this paper, we propose a generic, unsupervised way of sparse modeling of speech that does not require any inital knowledge about the senone class information. We take acoustic model outputs from the LF-MMI system (i.e., chain model) [5] and feed these features to shallow overcomplete autoencoder with ℓ_1-norm sparsity penalty on hidden unit activations. We study the recognition performance for the reconstructed acoustic model outputs. In addition, we conduct qualitative analysis of the hidden unit activations to investigate the modeling capacity of the aforementioned sparse autoencoder.

The paper is organized as follows. Section 2 provides the background and motivation for the present work. Section 3 details the databases and experimental setup. Section 4 presents the analysis of our findings. Finally, we conclude in Section 5.

This work was funded by the Swiss National Science Foundation under the project Sparse and HIerarchical Structures for Speech Modeling (SHISSM). e-mail: (see http://www.idiap.ch/en/people/directory).

2. BACKGROUND AND MOTIVATION

In this section, we present brief theoretical background for our work and our motivation.

2.1. Dictionary learning

Dictionary learning focuses on sparsity in the context of matrix factorization as an approximation $\mathbf{Y} \approx \mathbf{AX}$ of an observed matrix $\mathbf{Y}$ by the product of two unobserved matrices, $\mathbf{A}$ and $\mathbf{X}$. The goal of dictionary learning is to find both $\mathbf{A}$ and $\mathbf{X}$ that yield the sparsest representation of data $\mathbf{Y}$, subject to some approximation error ϵ

$$\min_{\mathbf{A},\mathbf{X}} \sum_{i=1}^{N} \|\mathbf{x}_i\|_0 \text{ subject to } \|\mathbf{Y} - \mathbf{AX}\|_2 \leq \epsilon, \qquad (1)$$

where $\mathbf{x}_i$ denotes the ith column of $\mathbf{X}$. In addition, since $\|\mathbf{x}_i\|_0$ is NP-hard, it can be relaxed to $\|\mathbf{x}_i\|_1$.

The joint optimization over $\mathbf{A}$ and $\mathbf{X}$ is non-convex. However, the problem can be solved as a convex objective function by following an iterative procedure where optimization for one variable is done while keeping the other variable fixed.

For our study, we take advantage of the insights from the online dictionary learning algorithm [6]. The algorithm alternates between steps of sparse coding [7] (i.e., estimating the sparse code with the current dictionary) and dictionary update (i.e., optimizing the dictionary using stochastic gradient descent).

The overcomplete dictionary $\mathbf{A}$ projects the data to a high-dimensional sparse space where the internal structures of the data are expected to be disentangled and only the relevant dimensions are activated. In other words, the state of discrimination between various data phenomena is increased and the blurry effect of variation on the data is degraded.

This property of dictionary learning may be helpful for speech recognition task where variation (from different sources like environment, speaker (gender, age, accent, pathological condition) etc.) constitutes challenges.

2.2. From dictionary learning to representation learning

We propose a novel sparse modeling framework based on the correspondence between the dictionary learning and neural networks shown in Fig. 2.

Our motivation is to turn the dictionary learning problem into a representation learning problem by means of autoencoders [8]. Thus, we take the columns in the dictionary (i.e., atoms) as the weights between the encoding layer and the output layer in autoencoder as shown in Fig. 2. The autoencoder training follows the same iterative process in the dictionary learning task. The forward pass can be seen as sparse coding with the current dictionary and the backward pass can be taken as dictionary update. The weights are updated according to the distance between the input and the reconstructed input.

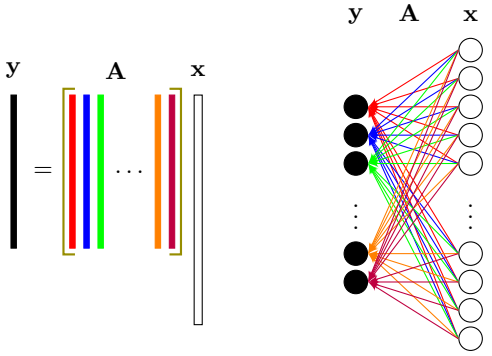


Fig. 2: The duality between the dictionary learning and autoencoders, note the colour-matching atoms-weights($\mathbf{A}$), input($\mathbf{y}$) and code representation($\mathbf{x}$). For simplicity, input and code are plotted in vector form.

2.3. Sparse autoencoders

The autoencoder has two main components, i.e., the encoder which maps the input to the code and the decoder which maps the code to the input reconstruction.

The overcomplete autoencoders are shallow autoencoders, with linear activation, tied weights (encoder weights are transpose of decoder weights), and no bias. In addition, we impose ℓ_1-norm penalty to enforce sparsity on hidden unit activations. The autoencoder is trained to minimize the reconstruction loss with an additional ℓ_1-norm penalty on the hidden unit activations as stated in Eq.2.

$$\mathbf{L} = \|\mathbf{y} - \mathbf{Ax}\|_2^2 + \lambda\|\mathbf{x}\|_1 \qquad (2)$$

The ℓ_1-norm penalty is weighted by the parameter λ which tunes the level of sparsity. This constraint is analogous to basis pursuit [9] in sparse recovery theory, and to LASSO regression [10] in statistics.

In short, we provide a high-dimensional representation space for the latent factors in the data to spread out, which has the same functionality with the overcomplete dictionary in [4]. However, in our approach, dictionary atoms (i.e., weights in neural network scenario) are learned together (not class-wise) in an unsupervised manner. After training, since only the relevant hidden units are expected to activate, feeding sparse overcomplete autoencoders with frames belonging to different senone classes implicitly result in different singular value decomposition operations. This can be seen as class-wise undercomplete dictionaries in [4] for low-rank modeling.

In this paper, the acoustic model outputs from the chain model system are fed to the sparse autoencoders. The output from the sparse autoencoder (i.e., reconstructed acoustic

model output features) are passed to the decoder to asses the recognition performance, whereas the hidden unit activations obtained from the sparse autoencoder are qualitatively analyzed for their speech modeling capacity.

3. EXPERIMENTAL SETUP

In this section, we introduce the datasets, the baseline systems and the sparse autoencoders used in our experiments.

3.1. Datasets

The experiments were conducted on the AMI corpus [11] which contains recordings of spontaneous conversations in meeting scenarios in English. The corpus provides audio recordings from close-talk (stated as IHM condition) and far-field (stated as SDM) microphones. Both close-talk and far-field speech streams have been recorded in parallel. The dataset is available at 16kHz sampling rate with nearly 100 hours of meeting recordings divided approximately as 81 hours train set, 9 hours dev and 9 hours eval set.

3.2. Baseline systems

Both baseline systems were implemented with the LF-MMI criteria [5] in Kaldi speech recognition toolkit [12]. The input for both systems was 40 dimensional high resolution MFCC features and 100 dimensional i-vectors. The output of the systems was the pseudo-log-likelihoods of size 3776, as the number of senones is 3776.

The configuration for the IHM acoustic model is presented in Fig. 3. The model configuration for SDM is same, except it does not contain the CNN and LSTM layers.

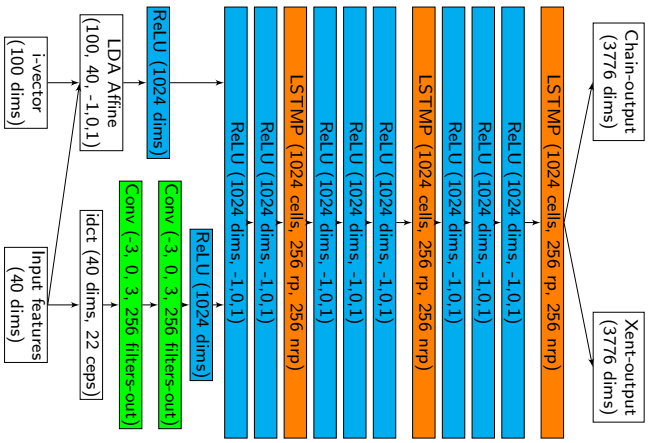


Fig. 3: The model configuration for IHM baseline. The green blocks represent CNN layers. The blue blocks represent TDNN layers with RELU activation function. The orange blocks denote LSTM layers. The xent-output layer is used for regularization purpose only. Figure adapted from [13].

3.3. Sparse autoencoders

Sparse autoencoders were implemented in Pytorch [14]. The input for the sparse autoencoder was the pseudo-log-likelihood vectors of size 3776 (chain-output layer in Fig. 3). We used overcomplete autoencoders with 11328 hidden units (three times the number of senones) in the encoding layer.

Thus, $\mathbf{y}$, $\mathbf{A}$ and $\mathbf{x}$ in Fig. 2 are a vector of pseudo-log-likelihoods with dimensionality of 3776, the decoder weights and the hidden unit activations with size of 11328, respectively.

Sparse autoencoders were trained with mean square error criterion using stochastic gradient descent with batch size 376 and learning rate 0.1. For λ, grid search was performed on [0.1-0.000001]. In addition, the default weight initialization scheme in Pytorch was utilized.

4. RESULTS AND ANALYSIS

In this section, we present the recognition performance for the reconstructed acoustic model outputs. In addition, we demonstrate the qualitative analysis of hidden unit activations to investigate the modeling capacity of the sparse autoencoder.

4.1. Recognition performance

The word error rates (WER) are 19.0% and 40.2% for IHM and SDM baseline systems respectively.

The output of the autoencoders were passed to the Kaldi decoder to estimate WER. As shown in Table 1, the results are comparable to the baseline system.

Architecture	IHM	SDM
Baseline	19.0	40.2
Overcomplete AE ($\lambda = 0$)	19.0	40.2
Sparse AE (λ=0.000001)	19.0	40.2
Sparse AE (λ=0.00001)	19.0	40.3
Sparse AE (λ=0.0001)	19.1	40.7

Table 1: The recognition performance (in WER%) for reconstructed acoustic model outputs compared to chain model baseline systems.

The LF-MMI baseline systems may be too adapted to AMI dataset, leaving no room for further improvement in the recognition performance. In addition, AMI may not be suitable for this task. In the future, we plan to test our method on more noisy datasets.

4.2. Qualitative analysis

In Section 4.2.1, we examine whether the sparse AE (λ=0.0001) could produce high dimensional sparse codes in the encoding layer. In Section 4.2.2, we examine whether the activation

patterns in high dimensional sparse activations could be useful for understanding low-dimensional senone subspaces.

4.2.1. *Sparsity of activations*

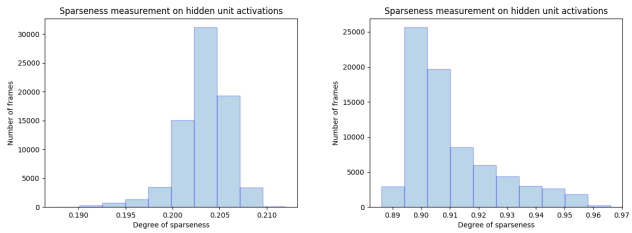
To measure the sparseness level of the activations, we used the following metric stated in [15].

$$\text{sparseness}\,(\mathbf{x}) = \frac{\sqrt{n} - (\sum |x_i|)\,/\sqrt{\sum x_i^2}}{\sqrt{n} - 1} \qquad (3)$$

where x is the hidden unit activations for one input frame (chain model outputs of size 3776 in our setup), n is the dimensionality of x (11328 in our setup). On the scale of zero to one, one represents the most sparse behaviour with only one non-zero component.

For LF-MMI pseudo-log-likelihoods on the development set, the sparseness level of the activations from overcomplete AE and sparse AE ($\lambda = 0.0001$) in Table 1 were examined.

Fig. 4 illustrates the sparseness behaviour of activations from autoencoders trained on IHM. For overcomplete AE (Fig. 4a), the sparseness level over all frames was below 0.21, whereas for sparse AE (Fig. 4b), it was above 0.89. Similarly, the autoencoders trained on SDM, for overcomplete AE, the sparseness level over all input frames was below 0.21, whereas the sparseness level for sparse AE was above 0.88.



(a) Overcomplete AE ($\lambda = 0$) (b) Sparse AE ($\lambda = 0.0001$)

Fig. 4: The histograms for the sparseness degree of the frame-wise activations. The distribution indicates that sparse AE produces sparse codes in the higher dimensional encoding layer.

4.2.2. *Senone subspaces*

To examine whether the active units in the encoding layer are relevant to the senone class, we took a subset of pseudo-log-likelihood frames from development set with frame-level senone class information. To approximate the response of the sparse AE ($\lambda = 0.0001$) to a specific senone class, we took the average of all activation vectors corresponding to the same senone class. Hence, we obtained the average activation vectors for each observed senone. We then binarized these senone-specific vectors by setting their mean as threshold. More specifically, the vector entries (i.e., dimensions) with the values below the designated threshold were set to 0 (i.e., off).

Senone ID	Phone	#Units	#∩	#∪	IoU score
642	/Z/	155	106	196	0.54
2463	/AW/	138	90	195	0.46
3589	/V/	157	93	211	0.44
1632	/W/	148	89	206	0.43
1434	/Z/	149	89	207	0.42
93	/D/	154	90	211	0.42
3412	/S/	144	87	204	0.42
3099	/Z/	146	87	206	0.42
403	/N/	160	91	216	0.41
417	/Z/	145	85	207	0.41

Table 2: The senone classes with top ten IoU score for senone 1147 with phone mapping /Z/ in IHM dataset. #Units is the number of active units out of 11328. $\#\cap$ is the number of common active units. $\#\cup$ is the number active units in the union set (logical or on binary activations). IoU is $\#\cap / \#\cup$.

In the end, for each senone class in our subset (976 for IHM and 929 for SDM out of 3776 senone classes), we had a binarized activation vector of size 11328 depicting the firing behaviour of hidden units.

To make the firing patterns interpretable, intersection over union (IoU) metric was used to find the senone classes with similar firing patterns. In addition, the phone mappings for the senones were found using Kaldi's *show-transitions* command.

The similar senones based on IoU scores were observed to have mappings for similar phones in IHM and SDM subset. For illustration (Table 2), we randomly selected senone 1147 from IHM. This senone has 147 active (i.e., not-zero, on) units out of 11328. With Kaldi, we have detected that it maps to phone /Z/. As a matter of fact, among ten closest senones shown in Table 2 (based on IoU score), four of them mapped to the same phone /Z/ as senone 1147.

In addition, based on the articulatory knowledge (i.e., the place and/or manner of articulation), phone /Z/ is close to /S/ (both are alveolar and fricative), /D/ (both are alveolar), /N/ (both are alveolar) and /V/ (both are fricative). On the other hand, /Z/ is not found to be relevant to /AW/ or /W/ based on the place and/or manner of articulation. Yet, these two phones are related to each other. This implies that senones which are close to each other (based on IoU score) also share similar linguistic information.

In other words, *if the units fired randomly, senones with similar firing patterns would not be related at all.* Hence, senones form their low-dimensional subpsaces in a common high dimensional space as shown in Fig. 1, with small number of common active units which are sensitive to phone level

information.

5. CONCLUSION

In this paper, we introduced our proposed approach for sparse modeling of speech, exploiting the connection We investigated the between dictionary learning and shallow overcomplete sparse autoencoders with ℓ_1-norm sparsity penalty on hidden unit activations. We investigated the impact of using reconstructed features obtained from the aforementioned sparse autoencoder in the context of speech recognition. And we yield comparable results. In addition, we qualitatively analyzed the hidden unit activations and demonstrated that sparse autoencoders were capable of producing distinctive and informative firing patterns, given the speech data.

6. REFERENCES

[1] K. Stevens, "Acoustic phonetics. current studies in linguistics (no. 30)," 1998.

[2] A. Jansen and P. Niyogi, "Intrinsic fourier analysis on the manifold of speech sounds," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*. IEEE, 2006, vol. 1.

[3] Y. Bengio et al., "Learning deep architectures for ai," *Foundations and trends® in Machine Learning*, pp. 1–127, 2009.

[4] P. Dighe, *Sparse and Low-rank Modeling for Automatic Speech Recognition*, Ph.D. thesis, 2019.

[5] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi," in *Interspeech*, 2016, pp. 2751–2755.

[6] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th International Conference on Machine Learning*. ACM, 2009, pp. 689–696, ACM Press.

[7] B. Olshausen and D. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607, 1996.

[8] M. Kramer, "Autoassociative neural networks," *Computers & chemical engineering*, vol. 16, no. 4, pp. 313–328, 1992.

[9] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, 1998.

[10] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[11] I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, et al., "The ami meeting corpus," in *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, 2005, vol. 88, p. 100.

[12] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011.

[13] Alim Misbullah, "Robust network structures for acoustic model on chime5 challenge dataset," in *Proc. CHiME 2018 Workshop on Speech Processing in Everyday Environments*, 2018.

[14] A. Paszke, S. Gross, S. Chintala, and G. Chanan, "Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration," 2017.

[15] P. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of machine learning research*, vol. 5, pp. 1457–1469, 2004.