A TWO-LAYER NON-NEGATIVE MATRIX FACTORIZATION MODEL FOR VOCABULARY DISCOVERY

Meng Sun, Hugo Van hamme

Department of Electrical Engineering-ESAT, Katholieke Universiteit Leuven,
Kasteelpark Arenberg 10, Bus 2441, B-3001 Leuven, Belgium

mengsun@esat.kuleuven.be, hugo.vanhamme@esat.kuleuven.be

ABSTRACT

A two-layer NMF model is proposed for vocabulary discovery. The model first extracts low-level vocabulary patterns based on a histogram of co-occurrences of Gaussians. Then latent units are discovered by spectral embedding of Gaussians at layer-1. Layer-2 discovers vocabulary patterns based on the histogram of co-occurrences of the latent units. Improvements in unordered word error rates are observed from the low-level representation to the two-layer model on the Aurora2/Clean database. The relation between the latent units and the states of an HMM is discussed.

Index Terms— non-negative matrix factorization, hidden Markov models, speech recognition

1. INTRODUCTION

Hidden Markov models (HMM) have been used successfully in automatic speech recognition (ASR) for several decades. Their success can be attributed to at least two aspects. One is modeling the observations of the hidden states with statistical models, e.g. Gaussian mixture models (GMM). The other is modeling the sequential nature of speech with a left-to-right structure. However, the model is also criticized for its strict left-to-right structure and first order memory. [1]

A new framework for discovering words in utterances and subsequently recognizing them was proposed in [2], where a histogram of acoustic co-occurrences (HAC) was used to represent speech, and non-negative matrix factorization (NMF) was used to extract recurring spoken patterns. As the acoustic co-occurrences can be defined with any time spacing, the HAC representation thus seems to be albe to model speech variations [3]. The HAC in [2] was based on a vector quantization (VQ) of a short term spectrum, which is a source of loss of accuracy. By replacing the code words with Gaussian posterior probabilities to generate HAC features (hence representing co-occurrences of Gaussians), one major difference with the HMM baseline is removed in this paper. A second difference that is addressed in this paper is the shallow structure of the HAC+NMF model, i.e. a word is characterized directly from its statistics of Gaussian posteriors. In contrast, an HMM recognizes a state level, where each state is in turn described in terms of a Gaussian mixture. In this paper, an intermediate abstraction level, comparable to an HMM state is introduced. The creation of the intermediate level (referred to as latent units) does not require supervision and is also obtained by a matrix factorization. Effectively, the co-occurrence modeling in HAC+NMF now happens at the level of the latent units.

The benefits of such an intermediate level are firstly that since there are less latent units than Gaussians, the co-occurrence statistics require less data to be estimated. This will result in an increased learning rate for new words, since the first layer (the relation between Gaussians and latent units) is reused. Secondly, the co-occurrence statistics can be constrained at the level of the latent units, e.g. using an upper-band-diagonal structure. While the original HAC+NMF model has a very weak description of sequential aspects (many sequences can map onto similar HAC representations), the constrained transitions of latent unit brings the proposed model closer to the sequential modeling of a left-to-right HMM, without imposing a rigid order which might lead to poor modeling of many pronunciation variants seen in spontaneous speech [3]. These arguments are listed to motivate the layered NMF model and are not the focus of this paper, where the primary question of accuracy of the model and analysis if the latent units is addressed.

The work in [4, 5, 6] shows that there is an intimate relation between an HMM and non-negative low-rank decompositions of co-occurrences. Internal variables (between Gaussians and states) and symmetric embedding matrices were assumed in [4, 6]. High order statistics were deployed to compute co-occurrences in [5] to ensure the HMM model is reconstructible from data. In this paper, we derive a matrix factorization model to discover latent units from the probabilistic relations of observations, Gaussians, and HMM states.

The primary goal of this paper is to improve the performance of vocabulary discovery by using the co-occurrences

The research was funded by the K.U.Leuven research grant OT/09/028(VASI).

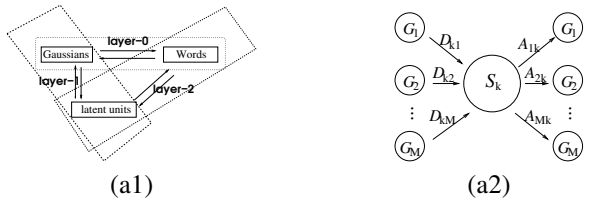
of the latent units as new HAC features with respect to the original model using the co-occurrences of Gaussians. The paper is organized as follows: the NMF model, embedding model and their algorithms are in section 2. The results are in section 3. The discussion and future work are in section 4.

Table 1. Summary of notation

G_i	Gaussian i
S_k	potential state or latent unit k
O_t	observation or feature vector of frame t
V_g	grounding matrix
V_a, X	data matrices of layer-0 and layer2
W_a, Y	acoustic pattern matrices of layer-0 and layer-2
W_g, Q	mapping matrices of layer-0 and layer-2
H, Z	coefficient matrices of layer-0 and layer-2
A, D	embedding matrices of out- and in- Gaussians
C, B	co-occurrence matrices of Gaussians and latent units

2. NMF MODEL FOR SPOKEN PATTERN DISCOVERY

The framework of the model is depicted in (a1) of Figure 1, where layer-0 discovers vocabulary patterns with co-occurrences of Gaussians, layer-1 embeds Gaussians into a number of latent units, and layer-2 extracts vocabulary patterns with co-occurrences of the latent units. The pattern discovery task in all layers is accomplished by matrix factorization. The two-layer model refers to layer-1 + layer-2, which makes a complete recognition system.



(a1) (a2)

Fig. 1. The model framework (a1) and the modeling of a latent unit (a2).

2.1. Layer-0: NMF based on the histogram of the co-occurrences of Gaussians

Given M Gaussians $\{G_i = \mathcal{N}(\mu_i, \Sigma_i), 1 \leq i \leq M\}$, the procedure to get the HAC representation is as follows.

- Gaussians and Posterior Probabilities
  First, the utterance is chopped up in overlapping signal analysis frames. For each frame, a MFCC+Δ+$\Delta\Delta$ vector O is computed. The likelihood of all Gaussians $p(O; G_i)$ is evaluated and the top-K_1 probabilities are retained and normalized to sum to unity to yield a sparse posterior distribution $p(G_i|O)$ of the frame.
- Co-occurrences of Gaussians
  The feature vectors O_{t_1} and O_{t_2} are said to form a *frame-pair* for a given *lag*-value if *lag*=$t_2 - t_1$. For each frame-pair, $K_1 * K_1$ *Gaussian-pairs* among $\{p(G_i|O_{t_1}) * p(G_j|O_{t_2})|i, j = 1, \ldots, M\}$ will have positive activation probabilities. For a given *lag*-value, the $((i-1) * M + j)$-th element of the M^2-dimensional HAC representation of an utterance is obtained by accumulating the activation scores of the Gaussian-pairs over all frame-pairs of an utterance, $\sum_{t_2-t_1=lag} p(G_i|O_{t_1}) * p(G_j|O_{t_2})$, which forms a column of the data matrix V_{lag}. We can take several different *lag*s to capture rich context dependency information and stack the data matrices to form $V_a = [V_{lag1}^T\ V_{lag2}^T\ V_{lag3}^T]^T$ (one column per utterance).

The grounding matrix V_g is used as supervision to associate acoustic representations with speech events and evidences: if the n-th utterance is known to contain the m-th vocabulary item l times, $V_{g_{mn}} = l$; otherwise, $V_{g_{mn}} = 0$. For details about the model and algorithms of layer-0 NMF, one can refer to [2]. The training model is as in Equation 1.

$$V \approx WH :\Leftrightarrow \begin{bmatrix} V_g \\ V_a \end{bmatrix} \approx \begin{bmatrix} W_g \\ W_a \end{bmatrix} H \tag{1}$$

For training NMF, by solving $\min_{W,H} C_0(V||WH)$ where C_0 is the extended Kullback-Leibler divergence [2], it yields the vocabulary patterns as the columns of W and the activations of the patterns as the columns of H. W_g reflects the relations between the acoustic parts W_a of the patterns and the vocabulary identities.

In the testing stage, we first estimate the activations H' of the acoustic patterns W_a by $\text{argmin}_{H'} C_0(V'_a||W_aH')$ on the test set. Then the activations of the vocabulary items is computed as $\hat{V}_g^{'} = W_gH'$. Since $\hat{V}_g^{'}$ indicates the presence of the vocabulary items in each utterance without ordering them in time, the performance metric that is adopted here is an *unordered* error rate. Suppose that the number D_u of different words occurring in the u-th test utterance is given, the D_u candidates with highest activation are retained in the u-th column of $\hat{V}_g^{'}$. The error rate is then defined as the sum of the number of incorrect digits (only substitutions), divided by the sum of D_u over the complete test set. Notice that the conventional word error rate can be used if NMF based recognition is used in a sliding window type of processing [2], but it involves all sorts of design choices that would blur the analysis.

2.2. Layer-1: embedding Gaussians to latent units

Suppose the observations are generated by an HMM, i.e. the latent units are HMM states. The configuration of the relations between the Gaussians and a prospective HMM state is

in (a2) of Figure 1, where G_i's are Gaussians, S_k is a state or a latent unit.

Notice there are two kinds of connections (weights) between Gaussians and states: A denotes the matrix of out-Gaussian probabilities $p(G_i|S_k)$, D denotes the matrix of in-Gaussian probabilities $p(G_j|S_l)$. Certainly, we can assume that $A = D^T$ as in the conventional HMM observation model or [6], but we have chosen not to impose the constraint in these first experiments. The relationship of the co-occurrences of Gaussians $p(G_i, G_j)$ (from G_i to G_j), the observation probabilities of out-Gaussians $p(G_i|S_k)$, the observation probabilities of in-Gaussians $p(G_j|S_l)$ and the co-occurrence probabilities between states $p(S_k, S_l)$ (from S_k to S_l) are described in Equation 2.

$$\begin{aligned} & p(G_i, G_j) \\ = & \sum_{k,l} \sum_{s,t} p(G_i|O_s)p(O_s|S_k)p(S_k, S_l)p(O_t|S_l)p(G_j|O_t) \\ = & \sum_{k,l} p(G_i|S_k)p(S_k, S_l)p(G_j|S_l) \end{aligned} \tag{2}$$

The matrix form is in Equation 3, where C is the co-occurrence matrix of Gaussians $C_{ij} = p(G_i, G_j)$, B is the co-occurrence matrix of the states $B_{kl} = p(S_k, S_l)$. As the number of Gaussians is usually larger than the number of states, the role of A or D is to *embed* the space of Gaussians to some latent space of states. Hence they are called *embedding matrices* and will be obtained by matrix factorization.

$$C = ABD \tag{3}$$

We call the prospective states *latent units* because the learning of Equation 3 is unsupervised. Thus the columns of A and the rows of D don't have to correspond to the underlying HMM states, but the two are expected to behave similarly. An algorithm with the normalization of columns of A and rows of D in Table 2 is used to learn A, B, D from the co-occurrences of Gaussians C by solving $\text{argmin}_{A,B,D}\, C_0(C||ABD)$ [7].

Table 2. Algorithms to get the embedding matrices

1	Initialization A,B,D,$n = 0$
2	While $n <$ # iterations
	(1) $P_{kl} \leftarrow \sum_i (A * B)_{il}, 1 \le k \le M$;
	(2) $D_{kl} \leftarrow D_{kl} * ((A * B)^T * (\frac{C.}{A*B*D}))_{kl} / P_{kl}$,
	(3) $D_{kl} \leftarrow D_{kl} / \sum_j D_{kj}$, $B_{tk} \leftarrow B_{tk} * \sum_j D_{kj}$;
	(4) $Q_{kl} \leftarrow \sum_i (B * D)_{ki}, 1 \le l \le M$;
	(5) $A_{kl} \leftarrow A_{kl} * ((\frac{C.}{A*B*D}) * (B * D)^T)_{kl} / Q_{kl}$,
	(6) $A_{kl} \leftarrow A_{kl} / \sum_i A_{il}$, $B_{lk} \leftarrow B_{lk} * \sum_i A_{il}$;
	(7) $B \leftarrow B. * (A^T * (\frac{C.}{A*B*D}) * D^T)$;
	(8) $n \leftarrow n + 1$;

With the model above, the embedding matrices A, D, the latent units and their co-occurrence matrix B for an underlying HMM are obtained. Now we prove that the co-occurrences of Gaussians of layer-0 are the estimates of the co-occurrences in Equation 2. The equation is in Equation 4 by summing away the S_k and S_l, where $p(O_s, O_t) = 1$ iff $t - s$=*lag*, so $p(G_i, G_j)$ is just the accumulated posterior probability of Gaussian-pairs in layer-0.

$$\begin{aligned} & p(G_i, G_j) \\ = & \sum_{k,l} \sum_{s,t} p(G_i|O_s)p(O_s|S_k)p(S_k, S_l)p(O_t|S_l)p(G_j|O_t) \\ = & \sum_{s,t} p(G_i|O_s)p(O_s, O_t)p(G_j|O_t) \end{aligned} \tag{4}$$

The rank and complexity of the embedding model grows linearly with the number of states of the underlying HMM and hence with the vocabulary size, so learning the latent units without further constraints has not been successful. Also, in [4, 5], HMMs of far smaller complexity are considered. However, the layer-0 model generates a per-word Gaussian co-occurrence model that is used to generate a set of embedding matrices as follows.

In the layer-0 model, the vocabulary patterns are stored in the columns of the acoustic pattern matrix W_a. With the grounding part W_g, we know which word a column of W_a represents. So for the first word $\text{argmax}_i W_{g_{i,1}}$, considering the acoustic part of the first column $W_{a:,1}$, taking the rows corresponding to the same *lag*, we can reshape it to a $M \times M$ matrix of co-occurrences of Gaussians C_1. Then by factorization $C_1 \approx A_1 B_1 D_1$, we obtain A_1, D_1 as embedding matrices of the Gaussians to the latent units and B_1 as the co-occurrences of the latent units of this word with contextual dependence *lag*. For the second word $\text{argmax}_i W_{g_{i,2}}$, the embedding matrix A_2, D_2 and the co-occurrences of the new latent units B_2 can be obtained in the same way from the co-occurrences of Gaussians C_2 reshaped from $W_{a:,2}$ with contextual dependence *lag*.

By making the same procedures for all the vocabulary patterns obtained in layer-0, we estimate the overall embedding matrices $A = [A_1, \ldots, A_L]$ and $D = [D_1; \ldots; D_L]$, and the overall co-occurrence matrix of latent units $B = \text{blkdiag}(B_1, \ldots, B_L)$. Different words have different units. For any other *lag*, the process is the same. One only needs to concatenate the obtained co-occurrences of latent units in the new data matrix X in section 2.3.

2.3. Layer-2: NMF based on the histogram of the co-occurrences of latent units

Now the utterances are going to be represented by the co-occurrences of latent units. With a fixed *lag*, an utterance is firstly represented by its histogram of co-occurrences of Gaussians, which is reshaped to be a M by M matrix C. Let K_2 be the number of Gaussians retained per frame. Then by using the obtained embedding matrices A, D of this *lag*, the co-occurrences of states B is estimated by the factorization $C \approx ABD$. Here only B is going to be updated with the algorithm of Table 2. Then B is reshaped back to a column vector as the representation of the utterance to be a column of the new data matrix X. Together with the grounding matrix

V_g, the training model is in Equation 5. At the end of the training stage, A,D,Y,Q are retained as key information for the recognition model.

$$\begin{bmatrix} V_g \\ X \end{bmatrix} = \begin{bmatrix} Q \\ Y \end{bmatrix} Z \tag{5}$$

In the testing stage, the Gaussian co-occurrences of a testing utterance is computed to be the matrix C'. By solving $C' \approx AB'D$, the co-occurrence matrix of the latent units B' is estimated, which is then flattened to be a column of the data matrix X' of the test set. By solving $X' \approx YZ'$ w.r.t. Z', and computing the activations of the digits by $\hat{V}_g^{'} = QZ'$, the unordered word error rates can be computed as in section 2.1.

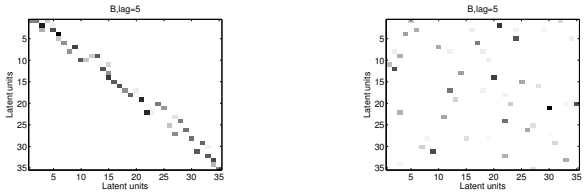
3. EXPERIMENTS AND RESULTS

The experiments were made on the Aurora2/Clean database which contains 8438 training utterances and 1001 test utterances of the 11 digits from male and female speakers. The window length for spectral analysis was 20ms and the frame shift was 10ms. The MFCC extraction used 30 Mel-filter banks from which 12 MFCC coefficients are computed plus the frame's log-energy. The three vectors of *Static, Velocity, Acceleration* were concatenated to a 39-dimensional feature vector on which a Gaussian mixture of M=3628 components were trained from HTK. No Gaussian was shared by any two HMM states. The word error rate of HMM in Table 3 was also obtained by using the Gaussians. For each frame, top K_1 and K_2 Gaussians with highest posterior probabilities were retained in layer-0 and layer-2 respectively. *lags* of 2, 5, and 9, so the time spacing of the frame-pairs were 20, 50 and 90 ms respectively, which represent contextual dependence with different time scales. R_1 is the number of latent units per digit to be discovered at layer-1. R=12 is the factorization dimension at layer-0 and layer-2. $R \geq$11 to ensure enough model complexity for the 11 digits.

Table 3. Unordered word error rates of the models

Model	K_1	K_2	lags	R_1	init. B	uWER
only layer-0	3	-	[2,5,9]	-	-	0.73%
three layers	3	3	[2,5,9]	35	rand	1.82%
only layer-0	5	-	[2,5,9]	-	-	0.58%
three layers	5	3	[2,5,9]	35	rand	0.44%
three layers	5	5	[2,5,9]	35	rand	0.44%
three layers	5	3	[2,5,9]	50	rand	0.47%
three layers	5	3	[2,5,9]	35	diag.	0.47%
HMM	-	-	-	16	-	0.15%

The unordered word error rates (uWER) are in Table 3. The uWER of the HMM is computed from the incorrectly decoded utterances by only considering the appearance or not of digits as in Section 2.1. The conventional word error rate of the HMM is 0.25%. Note that the HMM result is obtained with a different recognition paradigm: frame level Viterbi decoding instead of the utterance level co-occurrence count based detection approach explained in section 2.3.



(a1) (a2)

Fig. 2. The co-occurrences of the latent units of digit "one", R_1=35 units per digit, from W_a with K_1=5. (a1): upper-band-diagonal initialization. (a2): random initialization.

Experiments to initialize B with upper-band-diagonal structures were also tried. In the experiments, B was arranged as (from,to), which implies the lower triangle to be zero. The upper triangle should only have co-occurrences of "states" nearby. Thus the upper diagonal was narrow with *lag* diagonals. All other elements of B were zero. With the multiplicative updates used to solve NMF, they remain zero.

4. DISCUSSION

The two-layer model always performs better than the layer-0 model given K_1=5. But embedding matrices with K_1=3 fail to improve the corresponding layer-0 model. A sufficient number of Gaussians seem to be necessary to be retained per frame at layer-0 to model the relations between Gaussians and latent units (or potential HMM states). The performance of the two-layer model is robust to the number of latent units per digit, R_1, and the number of Gaussians retained per frame at layer-2, K_2.

Taking digit "one" with *lag*=5 as an example, the obtained co-occurrence matrices B's are shown in Figure 2, where (a1) is with upper-band-diagonal initialization and (a2) is with random initialization. In both cases, B is very sparse, showing that the latent units are sparsely co-occur as the HMM model suggests. The upper-band-diagonal and random initialization perform equally (within the experimental accuracy), showing that the two-layer model was able to discover the underlying sparse latent unit structure, which opens perspectives for modeling pronunciation variation.

The upper-band-diagonal initialization selects a permutation of the latent units by ordering them with "from"-"to" pairs. By analyzing the embedding matrices, we find that each latent unit usually activates Gaussians of several successive HMM states. So the units are indeed related to the HMM states, but there is (not surprisingly) not a one-to-one relation. Since the model can really discover an HMM-like structure, we may use its outputs as initializations to train an HMM or to make sequential decoding with its own sequential structure.

5. REFERENCES

[1] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", in *Proc. IEEE*, vol.77, no.2, pp.257-285, 1989.

[2] H. Van hamme, "HAC-models: a Novel Approach to Continuous Speech Recognition", in *Proc. International Conference on Spoken Language Processing*, pp. 2554-2557, Brisbane, Australia, 2008.

[3] M. Ostendorf, "Moving Beyond the 'Beads-On-A-String' Model of Speech", in *Proc. IEEE ASRU Workshop*, pp.79-84, 1999.

[4] B. Lakshminarayanan, R. Raich, "Non-negative matrix factorization for parameter estimation in hidden Markov models", in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pp.89-94, 2010.

[5] G. Cybenko, V. Crespi, "Learning hidden markov models using non-negative matrix factorization", *Technical report*, arXiv:0809.4086, 2008.

[6] B. Vanluyten, J. C. Willems, B. De Moor, "Structured Nonnegative Matrix Factorization with Applications to Hidden Markov Realization and Clustering", *Linear Algebra and its applications*, vol.429, no.1, pp.1409-1424, 2008.

[7] J. Yoo, S. Choi, "Probabilistic matrix tri-factorization", in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Taipei, Taiwan, 2009.