Non-negative Matrix Factorization for Word Acquisition from Multimodal Information Including Speech

Hugo Van hamme

K.U.Leuven, Dept. ESAT, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
hugo.vanhamme@esat.kuleuven.be

Abstract

The current generation of automatic speech recognizers incorporates a lot of hard coded knowledge about how speech is structured. Yet children seem to discover the structure of speech and language from examples. A new computational method to discover lexical items with little or no supervision, based on non-negative matrix factorization (NMF) of co-occurrence counts of low-level acoustic events is proposed and analyzed. It is shown how multiple information streams can be integrated and in particular that multimodal information relating to the message content facilitates vocabulary acquisition. A method to compute a phonetic interpretation of the models is given.

Index Terms: unsupervised learning, keyword spotting, co-occurrence data, non-negative matrix factorization, word acquisition, pattern discovery

1. Introduction

Billions of babies around the globe have succeeded to acquire a vocabulary of their native language. This is a process with a surprisingly small amount of direct supervision. Most words they know are not explained to them, but are learned from their significance in the world they live in. The task is even more complicated if we observe that most of the utterances we speak to our babies or children contain more than one word [1] and many words are never heard in isolation. Yet, they succeed finally in segmenting spoken utterances in words.

In this paper, we present a learning agent that discovers recurring acoustic patterns in speech, a problem that was addressed in a.o. [2], [3], [4], [5] and [6]. In this work, like in the present work, it is not assumed that an error-free transcription of the speech into symbols such as phones or even letters is available. In the DP-N-gram-approach [2], partial matches between two strings of phonetic transcriptions generated with an HMM-based phone transcriber are found using a dynamic programming algorithm allowing such that the matched transcriptions should be similar but not identical. Subsequently, an agglomerative clustering algorithm groups the similar phone strings into clusters that were then used for topic spotting. In [3], dynamic time warping at the acoustic level comparing MFCC vectors directly is used instead of matching through dynamic programming at symbolic level. The units that are most similar in this fault-tolerant match are retained as acoustic patterns. In the multigram approach of [4], a symbolically labeled input stream is entirely explained as a sequence a finite vocabulary of multigram models of variable length. Each of the vocabulary entries emits patterns of *variable* length according to a statistical model. Estimation and detection with multigrams are scrutinized and an extension to acoustic (non-symbolic) input using "temporal decomposition" is proposed. In [5], a cognitively more plausible of early lexical acquisition is proposed. A short-term recurrence filter finds similar patterns in automatically generated phonetic representations. But more importantly, the model links the acoustic pattern discovery with word-to-meaning acquisition by using multi-sensory input. The discovered structure is intermodal, i.e. the discovered speech patterns are linked to structure in other modalities.

A common property of the approaches discussed above is that models are aligned with the data and an explicit segmentation of the input patters is derived. In our approach, this will not be the case. Instead, a holistic representation of fixed dimension containing co-occurrence information will be formed at the utterance level. This representation is then decomposed in additive parts, which will represent the recurring patterns. Because we will explain all input utterances as closely as possible with a limited number of parts, these parts tend[1] to contain recurrent patterns in the input. It is counter-intuitive that segmentation is not a necessary step any more, nor for building the pattern models, nor for recognition of the patterns in unseen data. Yet, this is an important property of the proposed method and we radically do away with the 'beads-on-a-string' model [7]: sub-phones are combined to phones, which are linked together to words, and finally to sentences.

The core of the learning algorithm is proposed in this work is non-negative matrix factorization (NMF) [8]. By imposing non-negativity constraints, NMF allows only additive (not subtractive) combinations of parts, and thereby it is distinguished from other matrix factorization techniques such as principal component analysis or singular value decomposition.

Moreover, like in [5], learning will be crossmodal. In our setup, much like in the real world, this message is conveyed to the learning agent through other modalities than audition (such as vision). With the multimodal information the agent can link the recurring acoustic patterns with events in the other modalities. In this sense, the patterns are assigned a meaning and can therefore rightfully be called 'words'. In this respect, what is presented in this paper differs substantially from what we have presented earlier [6], [9]: exploiting the crossmodal information not only facilitates learning, it also augments the patterns with a meaning in the world the agent lives in.

The remainder of this paper is organized as follows: section 2 formalizes the problem that is addressed, section 3 explains how NMF is used to achieve the goals of crossmodal learning, section 4 provides experimental results which are analyzed in section 5 and extended to a larger vocabulary in section 6. We conclude with the discussion of section 7.

[1] With proper initialization and constraints, the parts will be forced to not necessarily model the most frequently occurring patterns.

2. Problem formulation

2.1. Multimodal information

The ACORNS project [10] aims to build a system that learns a vocabulary from multimodal information, much like a child does. Hence, the learning agent should be able to extract a vocabulary item like “ball” from spoken sentences such as “look at the ball” and “what a nice ball”. The relevance to the agent’s world of an utterance is conveyed through modalities other than audition. In this example, a round object could be presented through the visual channel simultaneously with the audio. However, in order to bypass the implementation of recognizers in modalities such as the visual or the tactile, it is assumed that the information of the other modalities can be represented by an unordered collection (a set) of tags drawn from a finite tag vocabulary. The tags idealize the input form other modalities and translate to the presence or absence of a vocabulary item (a keyword) in the audio stream. For the task complexity envisaged in the ACORNS project, the presence or absence of a tag suffices to represent the presence or absence of an object or action (transitive verbs are excluded) in the scene. Since the focus of this paper is on the *acoustic* aspects of pattern discovery, such an abstraction is not made for audition and hence every utterance u_j ($j = 1 \dots T$) is accompanied by a “semantic” description consisting of a set of K_j tags drawn from a set of cardinality L. In the simple tasks envisaged here, tags map to one or more words (e.g. “ball” or “look at”). Hence, the multimodal tags can be summarized in the $L \times T$ matrix $\mathbf{V}_w$ with

$$[\mathbf{V}_w]_{ij} \quad \begin{aligned} &= 1 \text{ if the } i\text{-the tag occurs in } u_j \\ &= 0 \text{ if it does not} \end{aligned} \tag{1}$$

The task of the *learning* algorithm is now to discover in the audio which acoustic patterns relate to each of the tags. The task of the *recognition* algorithm is to produce the correct set of tags given an unseen utterance (without tags).

2.2. Auditory preprocessing

The human ear is modeled with a MEL-scale filter bank whose log-outputs are sampled every 10 milliseconds. This is known to be a coarse, yet workable approximation. Subsequently, spectral changes are emphasized by adding first and second order time derivatives resulting in three data streams called *static* (S), *velocity* (V) and *acceleration* (A). The spectral similarity metric is Euclidean distance after cepstral transform of these three streams.

The input audio will be characterized by its similarity to examples in each of these streams. Therefore, the observed spectral vectors of each stream s (s = S, V or A) are clustered into N_s centroids using the K-means algorithm. The posterior probabilities $P_{i,s,n}$ of all centroids n now characterize any frame of audio data at frame (time) i in terms of its similarity to each of the centroids. Given the Euclidean distance metric used in clustering, each centroid can be represented by a Gaussian with spherical covariance. The posterior probabilities satisfy:

$$\sum_{n=1}^{N_s} P_{i,s,n} = 1 \text{ at frame } i \text{ and for } s = \text{S, V or A}$$

A special case is obtained in a vector quantization (or “winner takes all”) setting, where all posteriors are zero except for the centroid which is closest to the observation, which is assigned the value 1.

3. Unsupervised learning

3.1. Non-negative matrix factorization (NMF)

Given a matrix $\mathbf{V}$ of size $N \times T$ or non-negative real numbers, approximate non-negative matrix factorization rewrites this matrix as the product of factors $\mathbf{W}$ and $\mathbf{H}$ are of size $N \times R$ and $R \times T$ respectively and containing non-negative real numbers:

$$\mathbf{V} \approx \mathbf{W}\,\mathbf{H} \tag{2}$$

and $R \ll T$. The distance metric to measure the similarity of the left hand side and the right hand side of (2) is:

$$D(\mathbf{V} \,\|\, \mathbf{W}\mathbf{H}) = \sum_{i,j} \left(\mathbf{V}_{ij} \log \frac{\mathbf{V}_{ij}}{[\mathbf{W}\mathbf{H}]_{ij}} - \mathbf{V}_{ij} + [\mathbf{W}\mathbf{H}]_{ij} \right) \tag{3}$$

In this paper, the iterative multiplicative updates for minimizing (3) as outlined in [8] and reproduced in [6] are used. Multiplicative updates are easy to implement but suffer from slow convergence. $\mathbf{W}$ and $\mathbf{H}$ are initialized randomly and updated until the decrease in (3) drops below a threshold. Multiple random initializations are attempted and the result with minimal divergence is retained. In particular for the minimization of the Frobenius norm of the fitting error, a wide range of faster algorithms is described in literature. The main reason for our choice is the asymmetry in the divergence metric (3) and the property that in multiplicative updates, a zero matrix entry will always remain zero[1].

It is important to notice that in (2), $\mathbf{W}$ and $\mathbf{H}$ are not unique. For instance, scaling and permutation of the columns of $\mathbf{W}$ or the rows of $\mathbf{H}$ lead to equivalent solutions. Therefore, the columns of $\mathbf{W}$ will be normalized to sum to 1. The permutation problem will not hamper the analysis below.

3.2. Directed co-occurrence

The t-th utterance is represented by a single vector of lag-τ co-occurrence of acoustic events. In this paper, these acoustic events are the input nearing a centroid m (as defined in section 2.2). Hence, “lag-τ co-occurrence” signifies that the input nears a centroid m at time i while nearing centroid n at time $i+\tau$. Extension to other definitions of “acoustic event” is trivial. The co-occurrence is weighted with the (approximated) probability of the event, i.e. :

$$\left[\mathbf{C}_{s,t}^{\tau}\right]_{mn} = \sum_{i=1}^{I_t-\tau} P_{i,s,m} P_{i+\tau,s,n} \quad \text{with } m, n = 1 \dots N_s$$

where I_t is the length (in frames) of the t-th utterance. Notice that this co-occurrence is directed: $[\mathbf{C}_{s,t}^{\tau}]_{nm} \neq [\mathbf{C}_{s,t}^{\tau}]_{mn}$

Let $vec(\mathbf{C})$ denote the operator that stacks all columns of $\mathbf{C}$ in one column vector. The data matrix of a set of T utterances is now formed by

$$\mathbf{V}^{\tau} = \begin{bmatrix} vec\left(\mathbf{C}_{S,1}^{\tau}\right) & & vec\left(\mathbf{C}_{S,t}^{\tau}\right) & & vec\left(\mathbf{C}_{S,T}^{\tau}\right) \\ vec\left(\mathbf{C}_{V,1}^{\tau}\right) & \cdots & vec\left(\mathbf{C}_{V,t}^{\tau}\right) & \cdots & vec\left(\mathbf{C}_{V,T}^{\tau}\right) \\ vec\left(\mathbf{C}_{A,1}^{\tau}\right) & & vec\left(\mathbf{C}_{A,t}^{\tau}\right) & & vec\left(\mathbf{C}_{A,T}^{\tau}\right) \end{bmatrix} \tag{4}$$

[1] This property can also be implemented in additive updates by considering some entries as fixed (and zero).

Notice that all entries in the $(N_S^2 + N_V^2 + N_A^2) \times T$ matrix $\mathbf{V}^\tau$ are real and non-negative such that the tools of section 3.1 apply. In case vector quantization (see section 2.2) is used, $\mathbf{V}^\tau$ tends to be sparse.

3.3. A generative model

It is important to note that (4) is additive over time. If we imagine that an utterance is segmented in words, each word will contribute additively to the corresponding column of (4). Hence, if we place the co-occurrence counts of all words in a separate column of **W**, and if the corresponding rows of **H** would contain the presence of each word in each utterance, one would have

$$\mathbf{V}^\tau = \mathbf{W}\,\mathbf{H} \qquad (5)$$

For uniqueness, one can again renormalize the columns of **W** and inversely compensate the rows of **H**.

However, there are non-idealities:

- crossword co-occurrences of acoustic events depend on the sequence in which the words occur
- different realizations of a word might lead to different co-occurrence counts
- co-occurrence counts are proportional to the duration of the word

Therefore, equation (5) will not hold exactly and approximate NMF as in equation (2) is needed. Better models are even obtained if **W** also contains nuisance-columns, which are intended to model these non-idealities.

By mapping the columns of **W** to words, the question of the choice of R (the number of columns of **W**) is raised since the number of distinct words is unknown. However, since we assume each tag to correspond to at least one word, one should at least choose $R \geq L$.

Given the observation that a column of **W** can model different objects relevant to the structure of speech, the word “part” will be used to refer to a column this matrix.

3.4. Learning with NMF

In our problem, the words are unknown and NMF is used to separate out the words from the utterances. Though section 3.3 argues that a possible decomposition of co-occurrence count matrix in terms of parts corresponding to words is possible, it is not the only one. Actually, decomposition (2) approximates the data **V** as an additive combination of a limited number of parts. Utterances can be seen as a sequence of words, but also a sequence of phones, for instance, is a model that holds equally well. To conquer this ambiguity, constraints are imposed by exploiting the multimodal information (tags).

Eventually, we want to recognize the presence of tags in an utterance. **W** and **H** are therefore partitioned into

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_1 & \mathbf{W}_2 \end{bmatrix} \text{ and } \mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \end{bmatrix}$$

where $\mathbf{W}_1$ is a $N \times L$ matrix and $\mathbf{W}_2$ is a $N \times (R\text{-}L)$ matrix and similarly $\mathbf{H}_1$ has L rows. Each column of $\mathbf{W}_1$ will model a single tag. Since tags correspond to one or more words the argument of additive contributions of section 3.3 still holds and leads to a correct model for the co-occurrence data **V**. Matrix $\mathbf{W}_2$ contains parts that are not constrained except by non-negativity and the additive model (2): they could be anything from phones to multi-word expressions.

To implement the constraint on $\mathbf{W}_1$, we need to express that its i-th column only contributes if the i-th tag is associated with that utterance. Hence, we impose $[\mathbf{H}_1]_{it}$ to be nonzero if tag i occurs in utterance t by initializing $\mathbf{H}_1$ to zero when $\mathbf{V}_w$ is zero. Given the properties of multiplicative updates (section 3.1) the final **H** will retain this sparseness structure ensuring that the NMF decomposition only associates tag models in $\mathbf{W}_1$ to the utterances containing those tags. Without this constraint, the NMF tends to spend columns of **W** preferably on the more frequent acoustic patterns since this is most important to minimize the modeling error (3).

In earlier work [6] and [9] we have applied NMF to discover recurring acoustic patterns in speech. There are two major differences. First, the input data were phone lattices, which implies that the learning agent would have first discovered that speech is built up of phones that span up its auditory observation space space. In this work, we start directly form the acoustic events without phonetic knowledge. Secondly, we now deal with the multimodal information, which is important for learning by imposing constraints such that the parts model targeted information. Nevertheless, we observed in [6] that the discovered patterns modeled by columns of **W** could be mapped one-to-one to words.

3.5. Joint modeling of multiple streams

By construction, $\mathbf{V}^\tau$ contains co-occurrence data from the static, velocity and acceleration streams along its different rows. This idea can be extended further to include co-occurrence data at different lags. One may indeed expect that the time span over which acoustic events consistently co-occur depends on what one wants to model. For plosives, one might suggest a value around 10 ms for τ, while for modeling diphthongs or phone sequences, values in the order of 100ms seem more appropriate. Therefore, Q values of τ are included in the model:

$$\begin{bmatrix} \mathbf{V}^{\tau_1} \\ \vdots \\ \mathbf{V}^{\tau_Q} \end{bmatrix} \approx \begin{bmatrix} \mathbf{W}_1 & \mathbf{W}_2 \end{bmatrix} \mathbf{H} \qquad (6)$$

For these joint streams, the generative parts-based model still holds: the joint stream co-occurrences of utterances can be written as an additive combination of parts.

The capability of NMF to jointly model different streams of information is a surprising strength of the model that deserves further research. In the present case, these streams are synchronous (sampled at a 10 ms interval), but this is not a requirement. In section 5 for instance, segmental information will be incorporated.

Finally, though not essential to the success of the method, but for convenience in recognition, we add:

$$\mathbf{V} = \begin{bmatrix} \beta \mathbf{V}_w \\ \mathbf{V}^{\tau_1} \\ \vdots \\ \mathbf{V}^{\tau_Q} \end{bmatrix} \approx \begin{bmatrix} \multicolumn{2}{c}{\mathbf{W}_w} \\ \mathbf{W}_1 & \mathbf{W}_2 \end{bmatrix} \mathbf{H} \qquad (7)$$

where β is a positive real number and $\mathbf{W}_w$ is a real non-negative $L \times T$ matrix. Experiments show that β is not critical at all: the performance figures do not alter over a change of several orders of magnitude. The idea of adding additional rows to the data matrix extends further along the arguments given above for joint modeling of streams: $\mathbf{V}_w$ can be seen as a “semantic” information stream with one event per utterance. The matrix factorization (7) attempts to fit the acoustic and semantic information jointly.

The final training now consists of initializing $\mathbf{W}_w$ with random positive numbers along the diagonal, $\mathbf{W}_1$ and $\mathbf{W}_2$ with random positive numbers and **H** as explained in section 3.4.

3.6. Recognition with NMF

After training, $\mathbf{W}_w$, $\mathbf{W}_1$ and $\mathbf{W}_2$ are known. Given a set of T new test utterances (for which $\mathbf{V}_w$ is unknown), the matrix factor **H** is computed from the acoustic directed co-occurrences only using multiplicative updates, i.e. by minimizing the divergence (3) for model (6). Obviously, this matrix differs from the one obtained during training. Matrix factor **H** estimates to which extent the parts are present in the acoustic data. Since the full columns of **W** jointly fit the acoustic co-occurrence and $\beta\mathbf{V}_w$ on the training data, we can estimate the unknown $\mathbf{V}_w$ of the test data as $\mathbf{W}_w\mathbf{H}$. Finally, the activation of the tags can be computed using:

$$\mathbf{A} = \frac{1}{\beta}\mathbf{W}_w\mathbf{H} \tag{8}$$

The activations **A** (a $L \times T$ matrix) assume values between 0 and about 1. Entry $\mathbf{A}_{it}$ gives an estimate of the relevance of the i-th tag in the t-th test utterance. In the simple case where only one tag is assumed relevant to an utterance (see section 4.2), the recognized tags are the maximizers over the rows of **A**.

4. Experiments

4.1. The Dutch ACORNS database

Two male and two female adult speakers each utter 1000 sentences containing a single keyword in an infant-directed (IDS) and adult-directed (ADS) mode. Of these 7999 utterances, 1000 are used for testing, the rest is used for training. The keywords (which coincide with the tags), their translation into English and the number of occurrences in the training set are given in Table 1. In this first experiment, there is one and only one tag per utterance.

auto	car	696
bad	bath	696
boek	book	709
damian	proper name	176
flesje	bottle	708
isabel	proper name	178
luier	diaper	708
mama	mammy	697
mirjam	proper name	171
otto	proper name	175
papa	daddy	682
schoen	shoe	693
telefoon	telephone	712

Table 1: number of occurrences in the training set (right column) of each of the keywords (left column) and their translation (middle column).

Each of the keywords is embedded in a carrier sentence. The different carrier sentences are listed in Table 2. Here, <article> is either empty (for proper names), “*de*” (definite article for male or female nouns), “*het*” (neutral nouns) or “*een*” (indefinite article for all genders), while <key> is one of the keywords form Table 1. The keywords do not occur with an even distribution over in all carriers.

daar is <article> <key>	there is <article> <key>	709
(dag|hallo|hee|hoi) <key>	hello <key>	288
dat is <article> <key>	that is <article> <key>	1103
en daar komt <key>	and there comes <key>	138
en hier is <article> <key>	and here is <article> <key>	697
kijk <article> <key>	look <article> <key>	629
pak je <article> <key> ?	do you take <article> <key> ?	476
waar is <article> <key> nou ?	where is <article> <key> ?	712
wat een leuk(e) <key>	what a nice <key>	413
wijs <article> <key> aan	point at <article> <key>	631
zie je <article> <key> ?	do you see <article> <key> ?	694

Table 2: number of occurrences in the training set (right column) of each carrier phrase (left column) and its translation (middle column).

4.2. Training and recognition

In this experiment, we use a codebook for static, velocity and acceleration features of $N_S = 150$, $N_V = 150$ and $N_A = 100$ respectively with the vector quantization approach described in section 2.2. The codebook size is a compromise between quantization error and the size of the matrices to be handled. This value was an “educated guess” and not optimized on the present problem in any sense. The training as described in section 3.5 is performed with $R = 25$ and $\beta = 1000$ for one up to three values of the co-occurrence lag τ as listed in the first three columns of Table 3. Rows of **V** in equation (7) that are zero in the training are removed and the resulting number of rows N of **V** is given in column 4 of Table 3. The number of columns in **V** is always 6999, the number of training utterances. Subsequently, recognition of a single tag is performed using (6) and the resulting error rate on 1000 test utterances is listed in the rightmost column of Table 3.

We observe that most tags are correctly recognized despite the different speaking styles (IDS vs. ADS), the multiple speakers and the uneven occurrence frequency of the tags. Notice however that the training and test speakers are the same in this database, which is also most relevant to early language acquisition. It is remarkable to observe that the NMF-based recognition can successfully integrate correlated information streams. Given the small amount of errors, it is not possible to draw conclusions about the superiority of one configuration over another with any statistical significance.

τ_1	τ_2	τ_3	N	tag error rate
0	-	-	413	5.4 %
2	-	-	38986	0.9 %
5	-	-	48304	0.3 %
10	-	-	50731	0.6 %
2	5	-	87277	0.4 %
2	10	-	89702	0.3 %
5	10	-	99022	0.2 %
2	5	10	137995	0.2 %

Table 3: tag recognition results on the Dutch ACORNS database. The leftmost 3 columns give the co-occurrence lag in frames (multiple of 10 ms).

5. Examining the parts representation

By construction, a column of $\mathbf{W}_1$ successfully models co-occurrence of acoustic events that seem to be typical for a particular tag. However, since the number of carrier sentences is rather restricted and since the occurrence of the keywords in the tags is not evenly distributed, information in the carrier might contribute to the recognition of the tag. This would mean that the tag itself is not learned, but that the context it occurs in also plays a role. This is in itself not a negative property of a learning method: humans also exploit context and correlates to identify objects. A second question we would like to address in this section is what the model $\mathbf{W}_2$ contains. They could be words, multi-words, individual phones or any set of units that can parsimoniously model what is not covered by the tag models.

5.1. Analysis method

To make the parts models easier to interpret, we will add a phonetic stream to the model (7). This stream is not used for the learning process nor is it used during recognition since we do not want to assume a learning agent would dispose of knowledge of phonetics. We merely add it for analysis purposes. Like in [6], co-occurrence counts of adjacent phonetic events are used. Hereto, a phone lattice is generated with a phone recognizer [11] using an acoustic model for Dutch trained on 50 hours of read speech from the CGN corpus. For utterance t, the posterior probability $\gamma_{t,i}$ of the i-th arc in the lattice is computed according to [12] and the posterior probability $P_{t,k}$ of the k-th node is computed as well. The co-occurrence probability of phones m and n of adjacent arcs is accumulated over the lattice by summing:

$$\left[\mathbf{C}_t\right]_{mn} = \sum_{\substack{\text{arc } i \text{ with}\\ \phi(i)=m}} \; \sum_{\substack{\text{arc } j \text{ with}\\ \phi(j)=n}} \delta_{\omega(i),\alpha(j)} \frac{\gamma_{t,i}\,\gamma_{t,j}}{P_{t,\omega(i)}} \qquad (9)$$

where $\alpha(i)$ and $\omega(i)$ are the start and end node of the i-th arc and $\phi(i)$ is its phone identity and $\delta_{k,l}$ is Kronecker's delta. Finally, the phone co-occurrence counts of all T utterances are stacked in an observation matrix:

$$\mathbf{V}_P = \left[vec\left(\mathbf{C}_1\right) \;\; \cdots \;\; vec\left(\mathbf{C}_T\right) \right]$$

This phonetic information stream is now joined with that from the "acoustic space" (7) though the latter has a frame-synchronous event rate instead of the current segmental event rate. Subsequently, a NMF minimizing the divergence criterion is applied:

$$\mathbf{V} = \begin{bmatrix} \beta \mathbf{V}_w \\ \mathbf{V}^{\tau_1} \\ \vdots \\ \mathbf{V}^{\tau_Q} \\ \mathbf{V}_P \end{bmatrix} \approx \begin{bmatrix} & \mathbf{W}_w & \\ \mathbf{W}_1 & & \mathbf{W}_2 \\ & \mathbf{W}_P & \end{bmatrix} \mathbf{H} \qquad (10)$$

where $\mathbf{V}_P$ and $\mathbf{W}_P$ have an equal number of rows and the other matrices have the same dimensions as in (7). Moreover, $\mathbf{W}_w$, $\mathbf{W}_1$ and $\mathbf{W}_2$ are kept fixed to their values as obtained in section 4.2 during the multiplicative updates (which also removes the need for normalization of $\mathbf{W}$ and $\mathbf{H}$). With this constraint, we impose that the existing parts representation in the acoustic space is unchanged while the joint model also needs to explain the phonetic stream as well as possible. In other words, $\mathbf{W}_P$ hooks into the existing model and we associate a phonetic co-occurrence model to each tag model that was learned in the acoustic space before.

We can now measure the similarity of the parts learned by the NMF to a reference phonetic transcription. To this end, the phone string is first transformed to a chain of arcs, each with unit posterior probability. Then the reference phone co-occurrence count vector $\mathbf{c}_{\text{ref}} = vec(\mathbf{C}_{\text{ref}})$ is formed as in (9) for this particular lattice and normalized to sum to unity. The divergence measure between the reference transcription and the k-th column of $\mathbf{W}_P$ (also normalized to sum to unity) is now:

$$D\left(\mathbf{c}_{\text{ref}} \parallel \mathbf{W}_{:k}\right) = \sum_{i=1}^{N} \left(\left[\mathbf{c}_{\text{ref}}\right]_i \log \frac{\left[\mathbf{c}_{\text{ref}}\right]_i}{\mathbf{W}_{ik}} \right) \qquad (11)$$

A graphical representation of these divergences between $\mathbf{W}$-columns for the model with τ_1=2, τ_2=5 and τ_3=10 and the canonical transcription of each of the 37 words in the vocabulary is given in Figure 1. Here, each word is softly assigned to the $\mathbf{W}$-columns that respond most to it. The negative divergences (11) are exponentiated and normalized to sum to one over all candidate models. The resulting normalized divergence is a number between 0 (poor match) and 1 (canonical transcription with best match) where high values (white) indicate that the part ($\mathbf{W}$-column) responds most to the word.

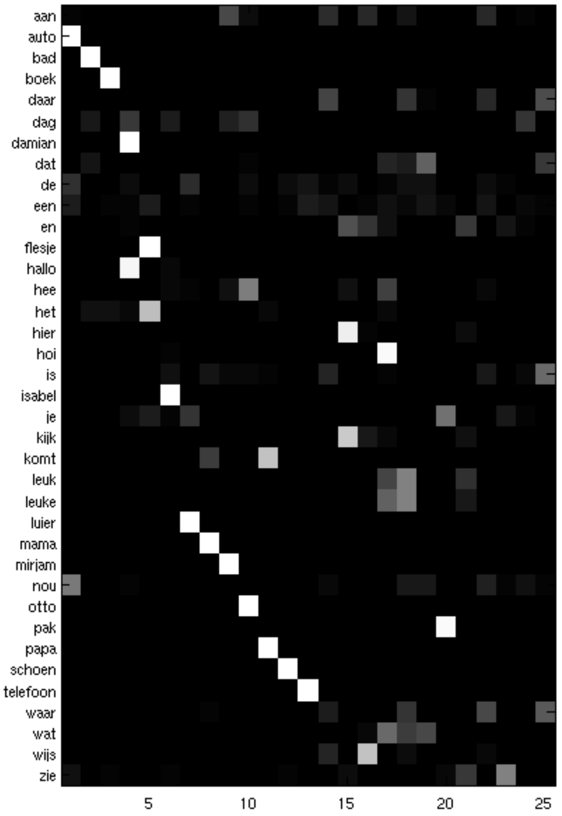

Figure 1: normalized divergence between canonical transcriptions (ordinate) and columns of $\mathbf{W}_P$ (abscissa). White represents close match.

5.2. The tag models

The first L (=13) columns of $\mathbf{W}$ model the tags by construction. We observe indeed from Figure 1 that each tag model responds well to one word from Table 1. However, some carrier words also produce good matches on some tag models. Acoustic similarity could be a reason for a good match, e.g. “auto” en “nou” are short words with a common vowel. However, this cannot be an argument for “komt” producing a good match on “mama” and “papa”. Closer inspection reveals that the carrier “en daar komt <key>” is only used with these two keywords and that “komt” is not used in any other context. Hence, the word “komt” helps to recognize these 2 tags. The association is not perfect, since “mama” and “papa” do occur in other carriers. Similarly, “dag”, “hallo” and “hee” are only used with the proper names which leads to an association effect. Finally, Dutch grammar imposes to use the definite articles “de” with “auto”, “luier”, “schoen” and “telefoon”, while “het” is used with “bad”, “boek” and “flesje”. Also these associations are apparent from Figure 1.

5.3. The carrier models

First notice that unlike the tag models, the carrier models are not essential for recognition (but improve performance). Since (7) is an approximate decomposition, differences between data and model can be tolerated. If recognition is attempted using only the L first columns of $\mathbf{W}$, tag error rates of 13% to 20% are observed, depending on the values of τ. Hence it is advantageous for recognition accuracy to not only rely on the tag models, but also try to explain the acoustic co-occurrences caused by the carrier sentences. Furthermore, as pointed out in section 3.3, parts in $\mathbf{W}_2$ can also account for model deviations such as co-occurrences of acoustic events that cross keyword boundaries.

The relation between $\mathbf{W}_2$-columns and carrier vocabulary is not that clear for multiple reasons. First, quite different factorizations can lead to local minima of the approximation error (3) that are nearly equal. This non-uniqueness could also be attributed to the fact that multiplicative updates for non-negative matrix factorization only guarantee non-increase of the divergence and may exhibit problematic convergence behavior [13]. But even in ideal cases where the global minimum is reached as in the example of Appendix A, the number of contexts in which the intuitively perceived parts occur must be large enough in order to be found as parts. This condition is definitely not satisfied in the present database design. These ambiguities occur because constraints of the type of section 3.5 are lacking fir the carriers. Secondly, the acoustic confusability of the words used in the carrier phrases blurs the divergence picture. Eight words contain only two phones and pairs like “*waar*”/“*daar*” or “*wat*”/“*dat*” are obviously acoustically similar. Third, the 12 parts (columns 14 through 25 of $\mathbf{W}$) spent on the carrier phrases do not suffice to model 24 words individually. Fourth, some words always occur together, such as the separable verb “*wijs aan*” and we cannot expect to find individual models for the components. It is therefore not possible to exactly match carrier models with words and multi-word expressions exactly, but an attempt is made in Table 4.

Column	word / MWE
14	“wijs ... aan” + “waar is” + “daar is”
15	“kijk” + “en hier”
16	“wijs ... aan”
17	“hoi” + “hee”
18	“wat ... leuk(e)”
19	“wat/dat”
20	“pak je”
21	“zie”
22	“waar”
23	“zie je”
24	“dag”
25	“waar is” + “daar is” + “dat is”

Table 4: matching columns of $\mathbf{W}$ (left) with words and multi-word expressions in the carriers (right).

6. Extended vocabulary

In section 5, there was only one tag per utterance. Most messages have more than one associated concept. In the theory explained above, no assumptions were made about the number of tags that can be associated to an utterance. In case of multiple tags, $\mathbf{V}_w$ will have more than one non-zero entry per column.

In the next experiment, we extend our tag set with “*daar*”, “*dat*”, “*hier*”, “*kijk*”, “*pak*”, “*waar*”, “*wijs*” en “*zie*” (all words for which visual, tactile or circumstantial cues can be given) so it now has cardinality 21 and repeat the training with the same choice for the co-occurrence lags and $R = 32$, as well as the analysis of section 5. The normalized divergence obtained between the canonical word transcriptions and this model is given in Figure 2.

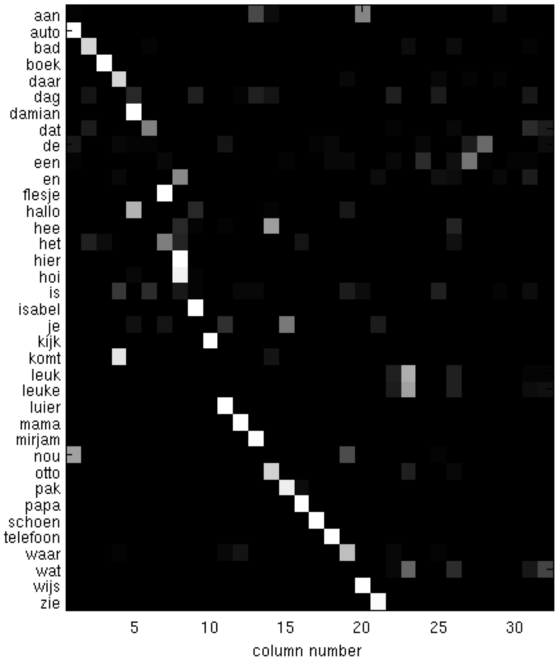

Figure 2: normalized divergence obtained with the extended tag set.

We observe that the new keywords that always occur with another word are grouped: "*daar*" + "*komt*" (column 4), "*pak*" + "*je*" (column 15), "*waar*" + "*is … nou*" (column 19), "*wijs*" + "*aan*" (column 20) and "*zie*" + "*je*" (column 21). There seems to be a better separation between the columns (remember that acoustic similarity also leads to a highlighted cell, e.g. "*je*" is a phonetic substring of "*luier*" and "*damian*"). For instance, the articles "*de*" and "*een*" do not seem to merge is as much with the keyword models and have received their own model in the carrier model $\mathbf{W}_2$. This also holds for "*je*", "*leuk(e)*", "*is*", "*wat*" have received a model that is better separated than before. Words like "*het*" are still mainly contained in the tag models.

In terms of tag recognition rate on the set of 13, slightly but statistically insignificant improvement over Table 3 is obtained, so further experimentation is needed to determine if a better parts segregation in the carrier models also leads to better tag recognition.

We conclude that the method can also handle multiple tags per utterance. The extended multimodal information helps to generate models of keywords and non-keywords that are less blurred.

7. Discussion and conclusions

This paper has shown that non-negative matrix factorization applied to co-occurrence matrices of acoustic events allows discovering recurring acoustic patterns with an associated meaning in a different modality. The multimodal information facilitates the separation of utterances into separate parts or models of words or multiple words. The parts are constructed such that acoustic and crossmodal information are best explained jointly. NMF successfully integrates multiple information streams that are not necessarily synchronous, can be strongly correlated and can have large dimensionality.

Multistream integration was used to compute a phonetic interpretation of the learned parts. It showed that the tag (keyword) models are not accounting for the tag only, but that words that often co-occur with keywords are also assimilated in the model.

A remarkable property of the NMF-based word models is that, unlike other unsupervised pattern discovery methods, a segmentation of the input in terms of the discovered units is not required. Instead, a holistic representation of the complete utterance (or more generally, of the analysis window) is made and approximately decomposed in parts. During recognition, the activity of all parts is computed where the order of the parts is not determined and the part boundaries are not aligned with the input. This also implies that a complete explanation of the input audio is not required, much like a human brain does not need to recognize all words in a message. The NMF-based recognition measures to which extent a particular model is present in the input, or equivalently, words are activated to a greater or lesser extent. Another property linked to the lack of segmentation is that the discovered patterns do not need to be contiguous, which allows to model separable verbs or exploit that words tend to occur together.

The lack of segmentation also has downsides. Though the directed co-occurrence representation is sensitive to the order in which the events occur, it does not impose a strict ordering of events (states) like in a HMM. The time dimension is only weakly present in the models. For instance, a cyclic permutation of a pattern will lead to the same co-occurrence statistics. Also, decoding word order and detecting word repetitions are not trivial. An extension of the present method to give a stronger account for time and order are on our research agenda.

8. Acknowledgements

This research was funded by the European Commission under contract FP6-034362 (ACORNS).

9. References

[1] Snow, C. and Ferguson, C. "Talking to Children: language input and acquisition", Cambridge: Cambridge University Press, 1977

[2] Nowell P. and Moore R.K., "The application of dynamic programming techniques to non-word based topic spotting", in *Proc. Eurospeech*, pages 1355-1358, Madrid, Spain, 1995

[3] Park, A. and Glass, J., "Towards unsupervised pattern discovery in speech," in *Proc. ASRU*, San Juan, Puerto Rico, 2005, pp. 53–58.

[4] Deligne S. and Bimbot F., "Inference of variable-length linguistic and acoustic units by multigrams," *Speech Communication*, vol. 23, no. 3, pp. 223–241, 1997.

[5] Roy, D. K. and Pentland A. K. "Learning words from sights and sounds: a computational model", *Cognitive Science* 26 (2002), 113–146

[6] Stouten, V., Demuynck, K. and Van hamme, H. "Discovering Phone Patterns in Spoken Utterances by Non-negative Matrix Factorisation." *IEEE Signal Processing Letters*, 2007. Accepted for publication.

[7] Ostendorf, M., "Moving beyond the 'beads-on-a-string' model of speech," in *Proc. ASRU* 1999, Keystone, Colorado, USA, Dec. 1999.

[8] Lee, D., and Seung, H., "Algorithms for non-negative matrix factorization," *Advances in Neural Information Processing Systems*, vol. 13, pp. 556–562, 2001.

[9] Stouten V., Demuynck, K., and Van hamme, H., "Automatically Learning the Units of Speech by Non-negative Matrix Factorisation." In *Proc. Interspeech*, pages 1937-1940, Antwerp, Belgium, August 2007.

[10] Boves, L., ten Bosch, L. and Moore, R. "ACORNS - towards computational modeling of communication and recognition skills", in *Proc. IEEE conference on cognitive informatics*, pages 349-356, August 2007

[11] Demuynck, K., Laureys, T., Van Compernolle, D., and Van hamme, H., "FLaVoR: a Flexible Architecture for LVCSR.", in *Proc. European Conference on Speech Communication and Technology*, pages 1973--1976, Geneva, Switzerland, September 2003.

[12] Wessel, F.; Schluter, R.; Macherey, K.; Ney, H., "Confidence measures for large vocabulary continuous speechrecognition", *IEEE Transactions on Speech and Audio Processing*, Volume 9, Issue 3, Mar 2001 pages 288 – 298

[13] Berry M.W., Browne M., Langville A.N., Pauca V. P. and Plemmons R.J., "Algorithms and applications for approximate nonnegative matrix factorizations", *Computational Statistics & Data Analysis* 52(1), (2007), pp. 155-173

Appendix A

In this section, we illustrate that valid decompositions into parts obtained with NMF can be counter-intuitive. The lesson to be learned is that the parts must occur in a “sufficiently large” number of contexts.

We consider the 7-segment alphanumeric display. We form a 9 × 5 grid of pixels that can be switched on or off. However, the 45 pixels are not controlled independently, but may lie on one of the 7 segments (if not, they are always switched off). We expect to learn that the image is composed of 7 segments.

We consider 3 training data sets: numeric, hexadecimal and alphanumeric symbols. For each symbol, a 35-dimensional column vector is constructed with a 0 or 1 in the row indicating if the pixel is on or off. These vectors are subsequently ranked in a matrix **V** of dimensions 35 × T, where T equals 10, 16 or 26 for the respective training sets. Subsequently, an NMF with common dimension $R = 7$ is computed. In all cases, the obtained divergence is 0, i.e. the factorization is without reconstruction error.

The decompositions are depicted in Figure 3 through Figure 5. Only for the 26 training tokens, we have enough independent data to successfully separate all segments. Since non-integer weights are allowed, a “3” in Figure 3 for example is formed as $\frac{2}{3}\,\mathbf{W}_{:5} + \frac{2}{3}\,\mathbf{W}_{:6} + \frac{1}{3}\,\mathbf{W}_{:7}$. These examples show that equivalent, but counter-intuitive solutions are still possible if the training data are insufficiently rich.

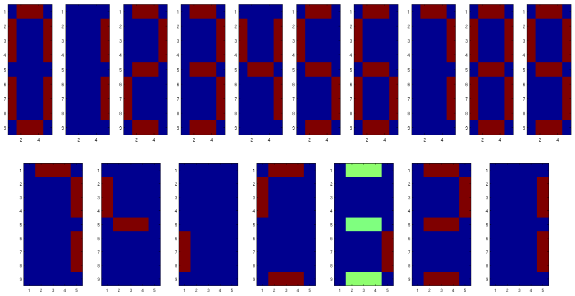

Figure 3: training data (top) and obtained parts **W** (bottom) for the numeric training set. Blue = 0, brown = 1 and green = 0.5

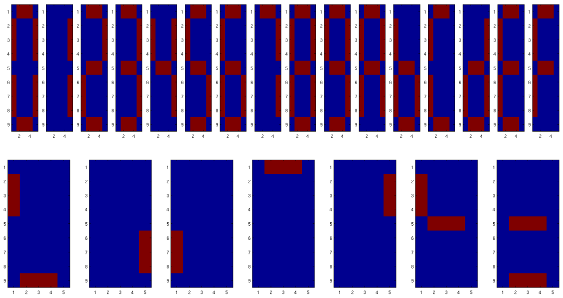

Figure 4: training data (top) and obtained parts **W** (bottom) for the hexadecimal training set. Blue = 0, brown = 1.

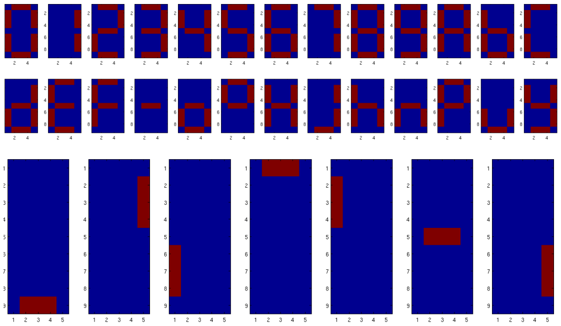

Figure 5: training data (top) and obtained parts **W** (bottom) for the alphanumeric training set. Blue = 0, brown = 1.