Project no. 034362

ACORNS

Acquisition of COmmunication and RecogNition Skills

Instrument: STREP
Thematic Priority: IST/FET

D4.1: Implementation and test of activation-verification mechanisms

Due date of deliverable: 30 November 2008
Actual submission date: 24 December 2008

Start date of project: 1 December 2006 Duration: 36 months

Project coordinator name: Prof. Lou Boves
Project coordinator organisation name: Radboud University,
Revision 1

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Implementation and test of activation-verification mechanisms

Joris Driesen, Hugo Van hamme

Katholieke Universiteit Leuven, dept. ESAT, Belgium
joris.driesen@esat.kuleuven.be, hugo.vanhamme@esat.kuleuven.be

Abstract and embedding in ACORNS

We describe a bottom-up, activation-based paradigm for continuous speech recognition. Speech is represented by co-occurrence statistics of acoustic events over an analysis window of variable length, leading to a vector representation of high but fixed dimension called "Histogram of Acoustic Co-occurrence" (HAC). During training, recurring acoustic patterns are discovered and associated to words through non-negative matrix factorization (NMF). During testing, word activations are computed from the HAC-representation and their time of occurrence is estimated. Hence, words in a continuous utterance can be detected, located and ordered. The plausibility of word activations is verified in multiple ways. First, the activations must exceed a threshold. Second, the locations (in time) of the detected words needs to be consistent over time. Third, we verify if the order in which words are activated corresponds to the expected activation patterns as learned through previous exposure to the language, a mechanism which is implemented as a second NMF-based learning and detection layer.

With this deliverable, we show that it is possible to acquire a vocabulary of 51 words using under supervision consisting of unordered utterance-level tags that simulate the multimodal input. Refer to D5.4.2 for experiments with weaker forms of supervision. Verification of detected words against knowledge sources such as a language model is important for scaling the vocabulary further towards the final goals of the project. The way in which we achieve this – through a second NMF layer – also proves that NMF-based learning can be cascaded, a technique that we want to apply at the lower, acoustic, level of speech processing. By introducing a layer below the current ones, reusability of acoustic representations and scalability towards a larger vocabulary size will be shown in the final year of ACORNS.

Index Terms: speech recognition, information discovery, non-negative matrix factorisation, co-occurrence statistics

1. Introduction

Hidden Markov Models (HMMs) have dominated automatic speech recognition (ASR) research for many decades. They have shown to be fairly adequate models for speech, but recent research has also uncovered some of their weaknesses such as poor robustness to pronunciation variation, co-articulation, assimilation and noise. Psycholinguistic studies on human speech recognition (HSR) 0 have also revealed differences with the way current ASR systems work. To deal with the intrinsic variation that is found in speech, ASR systems are based on statistical models that are learned from examples. However, the model structure is still hardwired and engineered: sentences are built of words, which are built of phonemes, which map to different allophones, which have statistical models. One of the goals of the ACORNS project 0 is to discover the structure in speech from data, much like a baby does not need linguistic theories to understand language. Although still outperformed by HMMs, this contribution presents encouraging results along this approach.

The HAC-model (histogram of acoustic co-occurrence) with its associated learning algorithm based on non-negative matrix factorisation (NMF) 0 is able to discover recurring acoustic patterns in speech without supervision 0 or with weak supervision 0. In the former case, the algorithm will identify which acoustic patterns reappear and therewith find the elements or latent structures that speech is composed of, and this without any guidance. With weak supervision, the utterances are accompanied by unordered information that relates to the spoken words, much like a baby receives, e.g., visual and tactile information that relates to the audio. This is a weaker form of supervision than is used in the training of HMM-based ASR systems, where utterances are described hierarchically in terms of word sequences down to HMM state sequences.

Once the HAC-models are trained, they can be used to decompose utterances in terms of the discovered latent structures, which will be words in this paper, but could also be phone-like units. This is a bottom-up

process: acoustic inputs activate words that compete, much like in the Shortlist model 0 of HSR. (There are differences with this model such as the absence of a pre-lexical level and "possible word constraints"). In this paper, HAC-models are extended to not only detect *which* words an utterance is composed of, but also in *which order* the words occur.

This paper is organized as follows: section 2 recaptures the ideas behind HAC-models, showing how words can be detected. In section 3, it is explained how the detected words can be ordered in time. Section 4 expands on the idea to build an activation-driven decoder. In section 5, the HAC-model is discussed and related to human speech recognition.

2. HAC-models

2.1. Histograms of acoustic co-occurrence

In the present bottom-up approach, recognition is driven by the co-occurrence of acoustic events. In general, these events are the occurrence of specific patterns in the time-frequency plane, ranging from patterns with a local time and frequency extent, such as the *patches* described in 0, to fullband patterns that span several 100ms like phones or even syllable-sized units. Specifically, in previous work 0, phones as well as vector-quantised (VQ) fullband spectra were used. The acoustic events are represented by discrete symbols from a set Σ, such as phone identities or VQ labels, their time of occurrence and an estimate of their posterior probability. In 0, acoustic events are represented as the edges in a directed acyclic graph, because it is then easy to define the distance or *lag* τ between edges α and β as the minimal number of vertices that need to be visited to travel from α to β, as well as the joint weight $p_{\alpha\beta}$ of the edge pair (α,β) as the sum of the posterior probabilities of partial paths starting with edge α and ending with edge β. An alternative *lag* measure could simply be the difference in time of occurrence of the acoustic events, which would eliminate the need to represent them in a lattice. However, for notational convenience, the lattice will be maintained below.

Next, all pairs of acoustic events $(A,B) \in \Sigma\times\Sigma$ are considered and the weight of all edge pairs with lag τ and carrying symbols A and B respectively is accumulated over the graph. Mathematically, the *n*-th utterance is characterized by

$$\mathbf{V}_{in} = \sum_{(\alpha,\beta)\in\Theta_i} p_{\alpha\beta} \qquad (1)$$

where Θ_i is the set of edge pairs with lag τ and labels A and B respectively. The index *i* is a one-to-one mapping of all combinations (A,B) to the integers $1 \ldots |\Sigma|^2$. In other words, a co-occurrence histogram of all possible acoustic event pairs is constructed. This results in a shift-invariant representation of fixed dimension (the square of the number of different symbols), independent of the length of the analysis window, which will be referred to as *HAC* (histogram of acoustic co-occurrence) or simply *histogram*.

In this paper, the acoustic events will be quantised fullband spectra computed at regular time instants or frames. The lattice of acoustic events then degenerates to a chain where each arc carries a VQ label with unity posterior probability. The HAC-representation is then the number of times all VQ symbol label pairs are observed τ frames apart.

For a given segment of speech, a unique high-dimensional HAC representation can be computed. Both the actual acoustic events occurring in the utterance as well as their order affect the HAC-representation. Conversely, a HAC representation does however *not* map to a unique symbol sequence or symbol lattice. For example, the histogram of the symbol sequence ABCDA and ABCDABCDA differ only by a scaling factor. The histograms of ABCDA and its (almost) cyclic permutation BCDAB are identical. In general, order is weakly represented in the HAC-model, much like a bigram language model only weakly represents grammar. Taking this analogy further, histograms over more than two symbols could alleviate this weakness at the cost of complexity. It will however be shown below that even with histograms of co-occurrence pairs, encouraging speech recognition results can be obtained. HAC-models are different from *convolutional NMF* 0, which describes traces in the acoustic space by an impulse response, a very rigid model for word-sized units.

2.2. Matrix factorisation for pattern discovery and recognition

If utterances are composed of one or more out of R recurring acoustic patterns such as words, the histograms have a linearity property that is essential in the proposed method: each such acoustic pattern is characterized by a HAC and hence the HAC of each utterance will be a (integer) linear combination of histograms. Define Θ_{ir} as the subset of Θ_i that originates from the r-th acoustic pattern. Then:

$$\mathbf{V}_{in} \approx \sum_{r=1}^{R} \sum_{(\alpha,\beta)\in\Theta_{ir}} p_{\alpha\beta} = \sum_{r=1}^{R} \mathbf{W}_{ir}\mathbf{H}_{rn}$$

where $\mathbf{H}_{rn}$ is the number of times pattern r occurs in utterance n and column $\mathbf{W}_{ir}$ is the histogram of acoustic co-occurrences for the r-th pattern. In matrix form:

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

Given their interpretation, all entries of $\mathbf{W}$ and $\mathbf{H}$ are constrained to be positive or zero. Because of these constraints, equation (2) is known as *Non-negative Matrix Factorisation* (NMF) 0. Since the observed symbols are subject to variability and uncertainty, equation (2) is only approximate and $\mathbf{W}$ and $\mathbf{H}$ are estimated as the positive matrices that minimize the divergence metric

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

An algorithm for finding $\mathbf{W}$ and $\mathbf{H}$ given $\mathbf{V}$ based on multiplicative updates is given in 0. This way, a small vocabulary can be discovered without supervision in a collection of utterances of continuously spoken words 0.

Without additional constraints, $\mathbf{W}$ (and therefore $\mathbf{H}$) is determined within a positive R-by-R scaling matrix $\mathbf{S}$ with a positive inverse: $\mathbf{W}\,\mathbf{H} = \mathbf{W}\,\mathbf{S}\,\mathbf{S}^{-1}\,\mathbf{H}$. The only possible such scaling matrices are products of diagonal matrices with a strictly positive diagonal (scaling of the columns of $\mathbf{W}$) and permutation matrices 0. To address scaling, the constraint that each column of $\mathbf{W}$ sums to unity is imposed, while permutation will not affect the results.

Once $\mathbf{W}$ is estimated on a training set, new utterances can be analysed with factorisation (2) by estimating $\mathbf{H}$, whose columns reveal the degree to which each discovered acoustic pattern is present in each new utterance.

2.3. Supervised learning

If it is known which words occur in each utterance, this information can be exploited to associate each column of $\mathbf{W}$ to a word identity. Therefore, the $M \times N$ *grounding* matrix $\mathbf{G}$ is formed, which holds in its m-th row and n-th column the number of times the m-th word occurs in the n-th utterance. Here, M is the number of word identities and N is the number of utterances available for supervised training. Subsequently, compute:

$$\begin{bmatrix}\mathbf{G}\\ \mathbf{V}\end{bmatrix} \approx \begin{bmatrix}\mathbf{W}_g\\ \mathbf{W}_v\end{bmatrix}\mathbf{H} \tag{4}$$

which expresses that word identity needs to be explained jointly with the acoustic data by common model activations $\mathbf{H}$. The common dimension R is chosen $R \geq M$. Experiments, not reported in this paper, have shown that overestimation of R is an acceptable strategy. The resulting columns of $\mathbf{W}_g$ will be zero, i.e., be associated with acoustic events that have no relevance to grounding, e.g., to a model for silence or filler words.

After supervised training, i.e., computing factorisation (4), recognition on unseen data is achieved by first computing $\hat{\mathbf{H}}$ in $\mathbf{V} \approx \mathbf{W}_v\,\hat{\mathbf{H}}$ using only the acoustic co-occurrence data and with fixed $\mathbf{W}_v$. The presence of words or their *activation* (i.e., an estimate of the grounding information: typically 1 for a word that is present and 0 if it is absent) in the test utterances is subsequently estimated as:

$$\hat{\mathbf{G}} = \mathbf{W}_g \hat{\mathbf{H}} \tag{5}$$

Notice an important difference with an HMM-based speech recogniser: each column of the matrix $\hat{\mathbf{G}}$ will reveal to which extent each trained word is present in the corresponding test utterance. However, it will say nothing about the *order* in which the words occur in the utterance, a problem that is addressed in the next section and forms the first novelty of this paper.

2.4. Activation-verification

In the previous section, it was shown that word activations can be computed. In this section, it is shown experimentally that this measure can be used to detect the absence or presence of words by thresholding the activation levels. This is a first mechanism of *activation-verification* (a second verification mechanism will be based on the estimated time of occurrence of the detected pattern - see section 4). In a level-based verification mechanism, only word candidates that have sufficient activation are retained for recognition or response generation. In this respect, they are equivalent to the *confidence levels* known in speech recognition, which are very useful for dialogue systems. In the present context, they are used for determining which words will be allowed in our *unordered* recognition result.

The level-based verification mechanism is evaluated on the TI-DIGITS corpus. The training data consist of 55 male and 57 female US-American adults, downsampled to 16 kHz. Since including examples of the acoustic patterns (words) in isolation would simplify the training problem, the isolated digit strings were removed from training and test, totaling 6159 connected digit sequences of length 2 through 7 for training.

For the acoustic information, 12 MFCC's plus log-energy are computed at a 100 Hz frame rate. A codebook of respectively 150, 150 and 100 for static, velocity and acceleration parameters is trained on the training set using the K-means algorithm. All training utterances are then processed resulting in a VQ-label for static, velocity and acceleration features per 10 ms analysis frame. Per utterance, the label co-occurrence histograms for the three streams are computed with a lag value $\tau = 5$, resulting in a 22500-dimensional vector for the static and the velocity stream and a 10000-dimensional vector for the acceleration stream. To improve discrimination between speech and non-speech, leading and trailing silence were stripped off each training utterance with an energy-based voice activation detector and considered as additional training utterances. Hence, a (very sparse) 55000 × 12318 data matrix $\mathbf{V}$ is obtained. The VQ histogram counts are divided by a fixed constant (100) such that the acoustic and grounding information have roughly the same weight in the cost function in supervised learning. Experiments have shown that the value of this constant is not critical: it can be changed over several orders of magnitude without significant impact. Subsequently, factorisation (4) is computed for $R = 12$ and $\mathbf{W}_v$ and $\mathbf{W}_g$ are stored for recognition.

To test the *verification* ability of the activation levels, the DET-curves are measured. These are based on two measures: the false alarm rate (i.e., the ratio of the number of false positives to the total number of negatives) and the miss rate (or 1-recall or 1-sensitivity or the ratio of the number of false negatives to the number of positives). Both error rates are plotted as *normal deviates*, such that the trade-off curves become more linear than in typical ROC-curves 0. In a first experiment, the false positives and negatives are measured at the utterance level, i.e., for each utterance of 2 to 7 digits (unknown to the detector), the activations of all 11 digits are computed and only those whose activation is above a threshold are considered as detected. The DET-curve is then obtained by varying the threshold. The DET curves for some digits are given in *Figure 1*. The other digits are removed from the plot so as to not overload it. The best results are obtained for "zero", while "oh" is (not surprisingly) the most difficult word to detect reliably. The DET-curve for all digits together (applying one common threshold) is shown in red. Overall, the detection performance is quite weak and could lead to a few percent of insertions (false alarms) and deletions (misses), depending on the chosen operating point.

The task of detecting the absence or presence of a word becomes more difficult as the length of the utterance increases. Therefore, the DET-curves are plotted for different utterance lengths in *Figure 2*. Clearly, shorter utterances yield a more reliable detection. Therefore, an activation-verification approach to speech recognition can possibly be optimized in a setting where the NMF model is used to detect words in a short window, rather than in a complete utterance. This will be discussed in section 4. In this approach, verification will also include information on *time*, which will now be discussed.

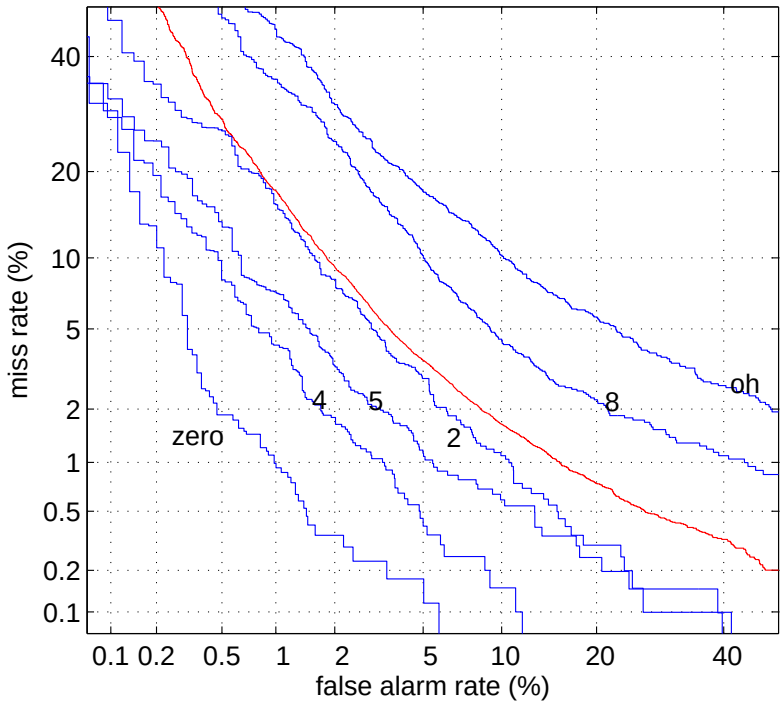

Figure 1: DET-curves of the digits "zero", "four", "five", "two", "eight" and "oh" (blue line) as well as the overall DET-curve for all 11 digits together (red curve, smooth and third from the top).

3. Word order and timing

The method of section 2 uses a "bag of words" approach, in that words can be discovered at training time, linked to word identities and subsequently can be spotted in given test utterance. However, word order, which is essential in language, is not modelled. A mechanism to find out in which order words occur in the test utterance is required. Apart from estimating word order, the estimated times of the detected words can also be used in an activation-verification framework as an additional verification criterion (see section 4)

3.1. Time-scaled histograms

Next to the histogram of acoustic co-occurrences, the edge pair weights are also multiplied with the time of occurrence of the first arc t_α (e.g., its ending time) and are accumulated over the lattice. Hence, with the notations of equation (1):

$$\mathbf{T}_{in} = \sum_{(\alpha,\beta)\in\Theta_i} t_\alpha p_{\alpha\beta} \qquad (6)$$

Using a similar reasoning as the one of section 2.2 for histograms, these *time-scaled* histograms (6) of an utterance are shown to be composed of the time-scaled histograms of acoustic patterns, but the weighting time is offset by the starting time of the pattern. Then:

$$\mathbf{T}_{in} = \sum_{j=1}^{R} \sum_{(\alpha,\beta)\in\Theta_{ij}} t_\alpha p_{\alpha\beta} = \sum_{r=1}^{R} \mathbf{X}_{ir}\mathbf{H}_{rn} + \sum_{r=1}^{R} \mathbf{W}_{v,ir}\mathbf{U}_{rn}$$

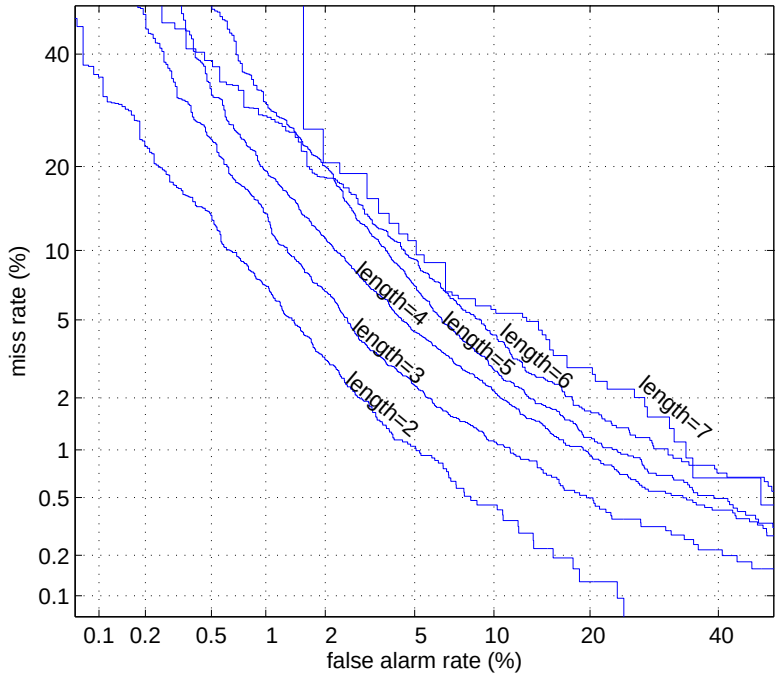

Figure 2: The impact of utterance length on the DET-curves.

(7)

where $\mathbf{X}_{ir}$ is the time-scaled histogram of the r-th pattern when aligned to some reference time and $\mathbf{U}_{rn}$ is the time offset of the r-th pattern in the n-th utterance. Hence, for supervised learning mode:

$$\begin{bmatrix}\mathbf{G}\\ \mathbf{V}\\ \mathbf{T}\end{bmatrix} \approx \begin{bmatrix}\mathbf{W}_g & \mathbf{0}\\ \mathbf{W}_v & \mathbf{0}\\ \mathbf{X} & \mathbf{W}_v\end{bmatrix}\begin{bmatrix}\mathbf{H}\\ \mathbf{U}\end{bmatrix} \tag{8}$$

The estimation of $\mathbf{X}$ in (8) appears to be a non-trivial problem. Only with knowledge of the time of occurrence of the words within the training utterances was an accurate estimate of $\mathbf{X}$ obtained. In order to avoid having to resort to exploiting additional (timing) information, the following approximation was considered: if the time-scaled histogram of the utterance is modelled as the sum of the time-scaled histograms of the acoustic patterns, but where event occurrence time t_α is approximated by the word occurrence time, $\mathbf{X}$ will equal $\mathbf{W}_v$ after normalisation. Accepting an offset in $\mathbf{U}$, (8) can be replaced by (supervised mode):

$$\begin{bmatrix}\mathbf{G}\\ \mathbf{V}\\ \mathbf{T}\end{bmatrix} \approx \begin{bmatrix}\mathbf{W}_g\mathbf{H}\\ \mathbf{W}_v\mathbf{H}\\ \mathbf{W}_v\mathbf{U}\end{bmatrix} \tag{9}$$

Rather than estimating $\mathbf{W}_g$, $\mathbf{W}_v$, $\mathbf{H}$ and $\mathbf{U}$ jointly on the training data using (9), slightly better results were obtained when training the $\mathbf{W}$-matrices from (4). This is not surprising since the approximation in the lower partition ($\mathbf{T}$) negatively influences the factorization of $\mathbf{V}$. During testing, $\mathbf{H}$ and $\mathbf{U}$ are estimated separately with fixed $\mathbf{W}_v$ and the time of occurrence of a pattern is found as $\mathbf{U}./\mathbf{H}$ (element-wise division).

3.2. Multiplicity

If a word occurs twice or more within an utterance, its number of occurrences as well as its times of occurrence will be added in one value for **H** and **U**. Hence, the estimated time of occurrence for the pattern will turn out to be an estimate of the "average" occurrence time of the pattern and makes ordering the words impossible. Although the multiplicity of a word can in principle be determined by inspecting **H** (or in case of supervised learning **G**), the patterns cannot be separated. This is intrinsically related to the property that HAC-models do not make an explicit segmentation of the data. The problem will be partially solved in section 4, where a method for handling disjoint repetitions of a word will be described.

3.3. Experiments

The setup and models described in section 2.4 are reused in this section. In addition, the matrix **T** is formed like **V**, but multiplying the counts data with the time of occurrence as in equation (6).

In this first evaluation, recognition and subsequently ordering of digits is attempted on a per-utterance basis. However, as outlined in section 3.2, it is not straightforward to locate multiple occurrences of the same word within a sentence, since (roughly) their average time of occurrence would be estimated. For consecutive repetitions of a digit, this is not a problem for order estimation. Therefore, utterances with non-adjacent repetitions of any digit were removed for testing, resulting in 4163 utterances. For example, "998" is retained, but "989" is removed as a test utterance. Given the number K_n of different digits occurring in the n-th test utterance, the K_n candidates with highest activation according to equation (5) are selected, yielding a word error rate of 2.83% and an *unordered* string error rate of 8.62%. Notice that this recognition result is unordered, so word error rate is defined as the sum of the number of incorrect digits that end up in the top K_n, divided by the sum of K_n over the complete test set. A string is incorrect if it contains any incorrect digit.

The recognised digits of each utterance are subsequently ordered by their estimated time of occurrence, i.e., **H** and **U** are estimated based on equation (9) and the digit position **U**./**H** is formed. For the n-th utterance, the recognised top K_n candidates are ordered according to their estimated time of occurrence. This yields an *ordered* string error rate of 11.72%. Hence, only in 3.1% of the strings did the ordering process introduce additional errors.

4. A sliding window decoder

The approach of section 3 takes a holistic approach to recognition in the sense that a complete utterance is analysed in terms of the components (words) it is composed of and subsequently those components are then located in time. At no point is there an attempt to break down the utterance into segments. For long utterances one can expect that *unmixing* the components leads to ill-conditioned cases.

In this section, a more local implementation of the same idea is examined. A sliding window of 400 ms is moved over the utterance in steps of 50 ms, word activation is computed and location of the best candidate is performed. This single best decoding strategy is viable only because no grammatical constraints need to be taken into account here. The candidate is accepted if its activation is higher than 0.25 and if its estimated location is within 40 ms of the window center. Examples of the activation levels and estimated times of accepted candidates by this process are shown in *Figure 3*. If in subsequent 50 ms frames the same word is detected, it is considered as a continuation of the same word and not as a repetition of it. However, when the same digit is repeated, this often – but not always – fails to produce an interruption in the activation of candidates, hence leading to deletion errors. Thus, the detection of adjacent repetitions of the same word is error-prone and not attempted in the sequel: adjacent repetitions of the same digit are mapped to a single occurrence in the reference and in the hypothesis during scoring.

This decoder is evaluated on a subset of the TI-DIGITS test set containing 6214 digit strings of length 2 through 7. As such, the algorithm leads to a word error rate of 7.40% (4.41% insertions, 2.19% deletions and 0.81% substitutions). It is observed that "oh" is often inserted after "zero", which is not unexpected, since the former is the last phone of the latter and the decoder does not have a constraint to find complete words. Invariantly, the word "oh" receives a large activation when the sliding window reaches the end of "zero". Therefore, all occurrences of "zero oh" were mapped to "zero". Similarly, "six" is often inserted before "seven" and "zero" and was only accepted in this context if its activation exceeds 0.5. This leads to a word error rate of 5.57% (2.56% insertions, 2.23% deletions and 0.78% substitutions). For comparison, a discrete density HMM was trained and tested on the same material, using the same VQ data and 7 states per digit (3

for silence). This yields 3.75% word error rate (1.53% insertions, 1.19% deletions and 1.02% substitutions). The lower performance of the HAC-model is mainly caused by insertions and deletions, which is not surprising given its extremely simple decoding strategy based on word activation without dynamic programming.

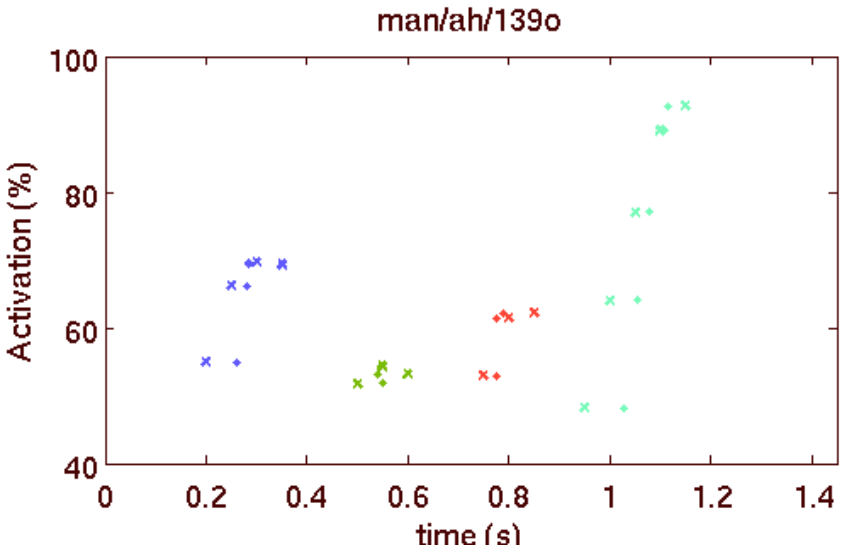

Figure 3: Activation level versus center position of analysis window (x) and estimated location (.) for the utterance "139oh". The activation of each digit is plotted using a different colour. The estimated positions cluster around four positions, one for each recognised digit.

5. Two-layered NMF

In section 4 we mentioned that acoustic similarity can activate unwanted words: the acoustic input "zero" can also activate "oh". When the goal is to recognise natural language, post-processing strategies like the one used in section 4 become unreasonable. In this section, we will describe how words can be activated by a first NMF layer that takes the acoustic evidence as input. A second NMF layer takes these word activation patterns over a time scale of hundreds of milliseconds and uses this to compute the final activations. The activation patterns are therefore verified against a new knowledge source: the language model. Apart from verifying the word activations in their context, the second layer can also learn that it is normal that similar words are both activated and learn the appropriate conclusion (i.e., in the context of section 4 to reject the activation of "oh" during "zero").

The second NMF layer is a verification step: its input and output are activation patterns of the same set of previously learned words. It therefore acts to accept or reject word activations. It learns what are normal activation patterns for a language and it also corrects the mistakes of the first layer. It is therefore expected to improve the recognition results of a single layer system.

To cast the behaviour of the second layer into an NMF problem, the activations in every utterance are converted to a vectorised representation. More precisely, for every possible combination of two multimodal tags, the product of their activations at a certain offset is calculated and summed over the entire utterance, as seen in equation (1) to form the HAC representation. Notice the difference with the HAC representation used in the first layer: now *weighted* activities are accumulated instead of integer counts. Using this representation of the activations, NMF can then easily be applied in the same way as in section 2.4.

5.1. Experiment

For this experiment it was decided to use the Dutch year 2 corpus of ACORNS 0. This corpus contains a total of 12109 usable utterances, spoken by 10 different speakers. Four of these speakers each provided about 2100 utterances. while the remaining six each provided 600. The ordering of the utterances was randomized and they were divided into a train set and a test set with a ratio of approximately 3 to 1, causing the train set to contain 9085 utterances and the test set 3024. The utterances contained simple sentences with

a maximum number of four keywords per sentence. In total, there are 51 different possible keywords consisting of both nouns and adjectives.

We define the layer 1 train set and the layer 2 train set, respectively containing 2/3 and 1/3 of the global train set, or in absolute numbers, 6056 and 3029 utterances, respectively.

The grounding matrices of the layer 1 train set and the test set are defined in the same way as in section 2.3. The HAC-representations of the data are also calculated, using time offsets of 20 ms, 50 ms and 90 ms. To obtain a finer acoustic granularity, the codebook size for acoustic vector quantization was increased with respect to earlier experiments with smaller vocabularies, to 150 labels for the static stream, 350 for the velocity stream and 200 for the acceleration stream. The codebook was trained on the entire database, using only speech frames selected by a straightforward energy-based voice activity detector. Making use of the acoustics matrix and the grounding matrix for the layer 1 train set, the internal representations W-matrix is trained. This training, as already mentioned above, happens in a weakly supervised way. The acoustic part of W is then used in the same way as in section 2.4: we apply a sliding window with a width of 400 ms and a window shift of 50 ms to the data in the layer 2 train and test set and calculate the sliding activations of the 51 different multimodal tags. These activations are then again converted to a matrix representation, using lags τ of 0, 2, 4, 6 and 8 window shifts of 50 ms, and combined with the grounding matrix to serve as input to the second layer of NMF. Specifically, with the activations for the layer 2 train set, a W-matrix is trained by which the activations of all multimodal tags in the utterances of the test set can be calculated. Much like

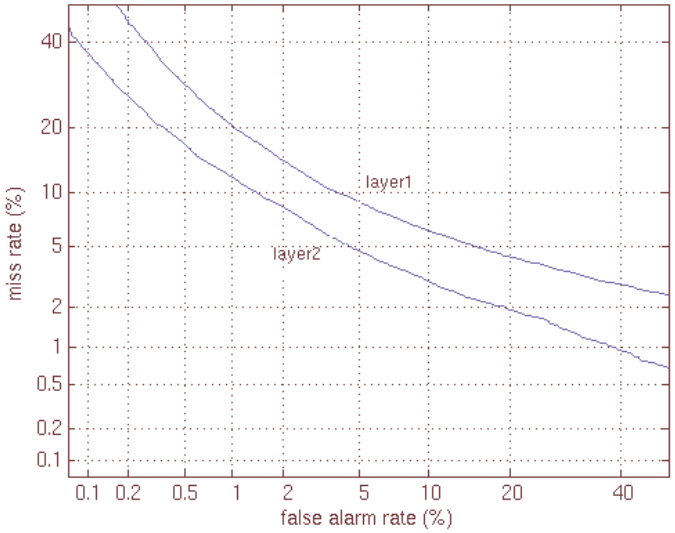

Figure 4: The DET-curves for detection by a common threshold applied to all 51 key words in the vocabulary for the first and the second NMF layer.

in section 2.4, the result can be shown using DET-curves. In *Figure 4* we show the result obtained with the first layer of NMF, compared to the result from the second layer. For both cases, the DET-curves are obtained by comparing the utterance-level activations of all 51 words for which we have learned an internal representation with the same threshold and accepting the word if the activation exceeds the threshold and rejecting it if it doesn't. The DET-curve of the first layer can be directly compared to the results in *Figure 1*, since they were obtained in essentially the same way. The lower level of performance in this experiment can be explained by the fact that the set of keywords to be detected is much larger, rendering the task more difficult. The figure also shows clearly that the performance of the second layer is superior to that of the first layer alone, proving the validity of this language-level verification mechanism.

The performance for individual keywords tends to vary. This is shown in *Figure 5* where the DET-curves for a small number of keywords are compared to that of the overall result (the smooth curve shown in red). In this figure, the activation of each word is compared to its own detection threshold.

Note that the division of the train set in a layer 1 train set and a layer 2 train set is absolutely necessary. If we were not to make this division, and used the same train set in layer 1 and 2, the activations used as input for the training phase in the second layer would be much cleaner than the activations used as input for the testing phase in the second layer, and would generate misleading results.

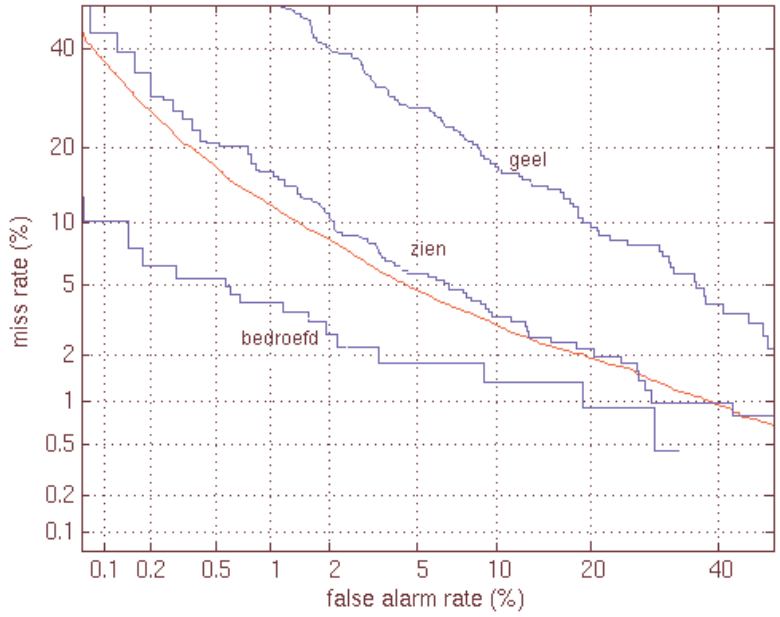

Figure 5: Layer-2 DET-curves for the keywords "geel" (yellow), "zien" (to see) and "bedroefd" (sad) shown in blue, together with the smooth overall DET-curve for all keywords combined, shown in red

6. Discussion and conclusions

The present approach shows some interesting similarities with models of human speech recognition (HSR). Most notable is that in HAC-models speech data are not segmented, but rather a window of speech is considered. Words are activated and compete much like is the case in the Shortlist model 0 of HSR. A holistic match of speech with high-dimensional models is made, which differs strongly from the approach taken by HMMs. The implicit segmentations that are generated in HMMs lead to sharp boundaries between words, a concept that is not so clear in HSR and that also might explain our insensitivity to strong cross-word coarticulations. However, HAC's failure to detect or even hypothesise word boundaries also leads to the problems with word multiplicity mentioned above. Progress on this front can probably be made if words are described as a sequence of subword units, which are then located and thus enabling the detection of word beginnings and endings. This will be explored in further research.

Notice also that at no point was order information in the training data used in any manner. At best, the presence/absence of words in the training data was used in supervised mode, though the NMF-based pattern discovery method can even function without supervision 0. But eventually, it is capable of recognising and ordering the discovered acoustic patterns, as shown above.

While NMF applied to the HAC-representation not only allows acquiring internal representations linking grounding information (tags) and acoustics, it also allows computing the *activations* of these internal representations from newly observed acoustic information. Words can be detected from these activations by a simple detection mechanism that consists of thresholding the activations with a common or word-

dependent value. While this detection process already forms a first *verification* mechanism, the plausibility of the word activations is also verified through other criteria. Accepted words should be consistently located at nearby time instants. Finally, we verify if the order in which words are activated corresponds to the expected activation patterns. These patterns are learned from example sentences, a mechanism which is implemented as a second NMF-based learning and detection layer.

The current implementation contains two layers of representations: one mapping the acoustic level directly to the lexical level and a second one for verifying word activations against the expectations built from prior exposure to the language. In ASR as well as in most models of HSR 0, 0, 0, 0, a pre-lexical (e.g., phonemic) level is also assumed. This level is required for building larger vocabularies, since re-use of acoustic representations (phonemes) can then be achieved for learning parsimonious lexical representations that require less storage and which can be learned from a small number of examples. A pre-lexical level, however, could also help decrease the ambiguity of the decoding process by imposing *possible word constraints* 0, 0. We consider further research into learning strategies in which such a layer emerges as the main goal of this work package in the final year of the ACORNS project.

7. References

Scharenborg, O., Norris,D., ten Bosch, L., McQueen, J.M., "How should a speech recognizer work ?", *Cognitive Science*, 29, 867-918, 2005.

Boves, L., ten Bosch, L., Moore, R.K., "ACORNS – Towards computational modeling of communication and recognitions skills", *IEEE Int. Conf. on Cognitive Informatics*, 349-356, 6-8 Aug. 2007

Lee, D.D., Seung, H.S., "Learning the parts of objects by non-negative matrix factorization", *Nature*, Vol. 401, No. 6755, 788-791, 1999

Stouten, V., Demuynck, K., Van hamme, H., "Discovering Phone Patterns in Spoken Utterances by Non-negative Matrix Factorisation." *IEEE Signal Processing Letters*, Vol. 15, 131-134, 2008.

Van hamme, H, "Integration of Asynchronous Knowledge Sources in a Novel Speech Recognition Framework", *ISCA ITRW workshop on Speech Analysis and Processing for Knowledge Discovery*, Aalborg, Denmark, June 2008, accepted.

Norris, D. "Shortlist: A connectionist model of continuous speech recognition", *Cognition*, 52, 189-234, 1994.

Ezzat, T., Bouvrie, J., Poggio, T., "Spectro-Temporal Analysis of Speech Using 2-D Gabor Filters", In. *Proc. Interspeech*, 506-509, Antwerp, Belgium, August 2007

Hans Laurberg, Mads Græsbøll Christensen, Mark D. Plumbley, Lars Kai Hansen, and Søren Holdt Jensen, "Theorems on Positive Data: On the Uniqueness of NMF", Comput Intell Neurosci. 2008; doi: 10.1155/2008/764206

Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M., "The DET Curve in Assessment of Detection Task Performance", In *Proc. Eurospeech* 1997, pp 1895-1898.

Ten Bosch, L. *ACORNS deliverable 5.4.2*

Smaragdis, P., "Non-negative Matrix Factor Deconvolution; Extraction of Multiple Sound Sources from Monophonic Inputs", In C.G. Puntonet and A. Prieto (Eds.), Independent Component Analysis and Blind Signal Separation, 494-499, Springer Berlin / Heidelberg, 2004

McClelland, J.L., Elman, J.L., The TRACE model of speech perception. Cognitive Psychology, 18, 1-86, 1996

Norris, D., "The Bayesian Reader: Explaining word recognition as an optimal Bayesian decision process." *Psychological Review*, 113(2), 327-357, 2006