LINEAR REGRESSION ON SPARSE FEATURES FOR SINGLE-CHANNEL SPEECH SEPARATION

Mikkel N. Schmidt and Rasmus K. Olsson

Technical University of Denmark
Richard Petersens Plads, Bldg. 321
DK-2800 Kgs. Lyngby, Denmark
Email: {mns,rko}@imm.dtu.dk

ABSTRACT

In this work we address the problem of separating multiple speakers from a single microphone recording. We formulate a linear regression model for estimating each speaker based on features derived from the mixture. The employed feature representation is a sparse, non-negative encoding of the speech mixture in terms of pre-learned speaker-dependent dictionaries. Previous work has shown that this feature representation by itself provides some degree of separation. We show that the performance is significantly improved when regression analysis is performed on the sparse, non-negative features, both compared to linear regression on spectral features and compared to separation based directly on the non-negative sparse features.

1. INTRODUCTION

The cocktail-party problem can be defined as that of isolating or recognizing speech from an individual speaker in the presence of interfering speakers. The ability of the human auditory system to solve this problem is impressive, even when using only one ear, or equivalently, listening to a mono recording of a mixture of different speakers. It is an interesting and currently unsolved research problem to devise an algorithm which can mimic this ability.

Different approaches for constructing such a system have been proposed, including methods based on computational auditory scene analysis (CASA) inspired by the mechanisms of the human auditory system; blind source separation (BSS) using little or no prior information about the signals; and machine learning methods, where speech models are learned from training data and subsequently used to separate the mixed speech. In this paper we focus on the machine learning approach, where isolated recordings of the individual speakers we wish to separate are available for training.

A number of such methods have been propsed. One approach, which arguably has been the most successful, is to use a hidden Markov model (HMM) based on a Gaussian mixture model (GMM) for each speech source and combine these in a factorial HMM to separate a mixture [1]. Direct inference in such a model is not practical because of the dimensionality of the combined state space of the factorial HMM. Roweis [1] shows how to obtain tractable inference by exploiting the fact that in a log-magnitude time-frequency representation, the sum of speech signals is well approximated by the maximum. Recently, impressive results have been achieved by Kristjansson et al. [2] who have devised an efficient method of inference that does not use the max-

approximation. In some situations, their system exceeds human performance in terms of the error rate in a word recognition task.

Another class of algorithms, here denoted 'dictionary methods', generally rely on learning a matrix factorization, in terms of a dictionary and its encoding for each speaker, from training data. The dictionary is a source dependent basis, and the method relies on the dictionaries of the sources in the mixture being sufficiently different. Separation of a mixture is obtained by computing the combined encoding using the concatenation of the source dictionaries. As opposed to the HMM/GMM based methods, this does not require a combinatorial search and leads to faster inference. Different matrix factorization methods can be conceived based on various a priori assumptions. For instance, independent component analysis and sparse decomposition, where the encoding is assumed to be sparsely distributed, have been proposed for single-channel speech separation [3, 4]. Another way to constrain the matrices is achieved through the assumption of non-negativity [5, 6], which is especially relevant when modeling speech in a magnitude spectrogram representation. Sparsity and non-negativity priors have been combined in sparse, non-negative matrix factorization [7] and applied to music and speech separation tasks [8, 9, 10].

In this work, we formulate a linear regression model for separating a mixture of speech signals based on features derived from a time-frequency representation of the speech. As a set of features, we use the encodings pertaining to dictionaries learned for each speaker using sparse, non-negative matrix factorization. We evaluate the performance of the method on synthetic speech mixtures by computing the signal-to-error ratio, which is the simplest, arguably sufficient, quality measure [11].

2. METHODOLOGY

The problem is to estimate P speech sources from a single microphone recording,

$$y(t) = \sum_{i=1}^{P} y_i(t), \tag{1}$$

where $y(t)$ and $y_i(t)$ are the time-domain mixture and source signals respectively.

We compute the separation in a time-frequency magnitude representation, $\mathbf{Y} = \mathrm{TF}\{y(t)\}$, where $\mathbf{Y}$ is a non-negative real-valued matrix with spectral vectors as columns, i.e., we do not try to estimate the phase. Instead, to compute the separated time-domain signals, we refilter the original mixture signal using the estimated magnitude spectra.

2.1. Linear regression

To perform the separation we propose a simple method, namely linear regression. We estimate the magnitude time-frequency representations of the sources in a mixture as a linear regression on features derived from the mixture. The linear model reads,

$$Y_i = W_i^\top (X - \mu 1^\top) + m_i 1^\top + N, \qquad (2)$$

where $Y_i = \text{TF}\{y_i(t)\}$ is the time-frequency representation of the i'th source, W_i is a matrix of weights, X is a feature matrix derived from Y; in the following we discuss these features in detail. The vectors μ and m_i are the means of the features and the sources respectively and are computed on training data. The matrix N is an additive noise term.

We make two assumptions in order to obtain a particularly simple maximum a posteriori (MAP) estimator based on this model: i) the noise is zero mean normal i.i.d. with variance σ_n^2 and ii) the prior distribution of the weights is zero mean normal i.i.d. with variance σ_w^2. For a detailed derivation of the MAP estimator, see e.g. Rasmussen and Williams [12]. Under these assumptions, the MAP estimator of the i'th source is given by

$$\hat{Y}_i^* = \Gamma_i \Sigma^{-1} (X^* - \mu 1^\top) + m_i 1^\top, \qquad (3)$$

where X^* is the feature matrix computed from the test mixture, Y^*, and

$$\Gamma_i = (Y_i - m_i 1^\top)(X - \mu 1^\top)^\top, \qquad (4)$$

$$\Sigma = (X - \mu 1^\top)(X - \mu 1^\top)^\top + \frac{\sigma_n^2}{\sigma_w^2} I. \qquad (5)$$

Here, X is a matrix with feature vectors computed on a training mixture and Y_i is the corresponding time-frequency representation of the source.

When an isolated recording, Y_i is available as training data for each of the speakers, it is necessary to construct the training feature matrix, X, from synthetic mixtures. One way to exploit the available data would be to generate mixtures, X, such that all possible combinations of time-indices are represented. However, the number of sources and/or the number of available time-frames would be prohibitively large. For example, the five minute training data used for each speaker in this paper lead to matrices Y_i with approximately 10^4 columns. Creating all combinations of just two speakers would require computing a feature matrix, X, having 10^8 columns.

A feasible approximation can be found in the limit of a large training set by making two additional assumptions: i) the features are additive, $X = \sum_i^P X_i$ with mean vectors μ_i, which is reasonable for, e.g., sparse features, and ii) the features are uncorrelated between sources such that all cross-products are negligible. Then, we can make the following approximation

$$\Gamma_i \approx (Y_i - m_i 1^\top)(X_i - \mu_i 1^\top)^\top, \qquad (6)$$

$$\Sigma \approx \sum_{i=1}^P (X_i - \mu_i 1^\top)(X_i - \mu_i 1^\top)^\top, \qquad (7)$$

which allows us to use isolated recordings of each source as training data directly without generating synthetic mixtures.

2.2. Features

In this work, we explore two sets of feature mappings. The first, and most simple, is to use the mixture time-frequency representation itself as input to the linear model, $X_i = Y_i$, $X^* = Y^*$. With these features, the spectra of each speaker is modeled as a linear combination of the mixed speech spectra; this allows the model to capture correlations between frequency bands specific to each speaker.

The second feature set we explore is the encodings of a sparse, non-negative matrix factorization algorithm (SNMF) [7]. Possibly, other dictionary methods provide equally viable features. In the SNMF method, the time-frequency representation of the i'th source is modelled as $Y_i \approx D_i H_i$ where D_i is a dictionary matrix containing a set of spectral basis vectors, and H_i is an encoding which describes the amplitude of each basis vector at each time point. In order to use the method to compute features for a mixture, a dictionary matrix is first learned separately on a training set for each of the sources. Next, the mixture and the training data is mapped onto the concatenated dictionaries of the sources,

$$Y_i \approx D_i H_i, \qquad Y^* \approx D H^*, \qquad (8)$$

where $D = [D_1, \ldots, D_P]$. The encoding matrices, H_i and H^*, are then the features used as input to the linear model, $X_i = H_i$, $X^* = H^*$.

In previous work, the sources were estimated directly from these features as $\hat{Y}_i^* = D_i H_i^*$ [10]. For comparison, we include this method in our evaluations. This method yields very good results when the sources, and thus the dictionaries, are sufficiently different from each other. In practice, however, this will not always be the case. In the factorization of the mixture, D_1 may not only encode Y_1 but also Y_2 etc. This suggests that the encodings should rather be used as features in an estimator for each source.

3. EVALUATION

The proposed speech separation method was evaluated on a subset of the GRID speech corpus [13] consisting of the first 4 male and first 4 female speakers (no. 1, 2, 3, 4, 5, 7, 11, and 15). The data was preprocessed by concatenating 5 minutes of speech from each speaker and resampling to 8 kHz. As a measure of performance, the signal-to-error ratio (SER) averaged across sources was computed in the time-domain. The testing was performed on a total of 9 minutes of synthetic 0 dB mixtures of two speakers, constructed using all combinations of speakers in the test set.

The time-frequency representation of the sources and mixtures were computed by normalizing the time-signals to unit power and computing the short-time Fourier transform (STFT) using 64 ms Hamming windows with 50% overlap. The absolute value of the STFT was then mapped onto a mel frequency scale using a publicly available toolbox [14] in order to reduce the dimensionality. Finally, the mel-frequency magnitude spectrogram was amplitude-compressed by exponentiating to the power p. By cross-validation we found that best results were obtained at $p = 0.55$ which gave significantly better results compared with, e.g., operating in the amplitude ($p = 1$) or the power ($p = 2$) domains (see Figure 4). Curiously, this is similar to the empirically determined $p \approx 0.67$ exponent used in power law modelling of perceived loudness in humans, known as Stevens' Law (see for example Hermansky [15]).

When learning the sparse dictionaries, the SNMF algorithm was allowed 250 iterations to converge from random initial conditions drawn from a uniform distribution on the unit interval. The number of dictionary atoms was fixed at 200. The SNMF method has a sparsity parameter, λ, which we chose by cross-validation to $\lambda = 0.15$. When computing the encodings on the test mixtures, we did not enforce sparsity, as cross-validation showed that best results were obtained at $\lambda = 0$.

Since the methods separate speakers in the magnitude time-frequency domain and do not estimate the phase of the separated signals, we used a simple refiltering method to compute separated time-domain signals. We computed the STFT of the mixture signal and performed a binary masking and subsequent inversion as described by Wang and Brown [16]. Audio examples of the reconstructed speech are available online [17].

In Figures 1 and 2, the performance is shown for the different methods. The acronyms MAP-Mel and MAP-SNMF refer to using the mel spectrum or the SNMF encoding as features in the linear regression, respectively. For reference, results are provided for the basic SNMF approach as well [10].

We also experimented with using a stacked feature representation, where five consecutive feature vectors spaced 32 ms apart were combined into one large feature vector as a simple means to modeling temporal dynamics. In the figures, this is denoted by the suffix "5".

The best performance is achieved for MAP-SNMF-5, reaching an $\simeq 1.2$ dB average improvement over the SNMF algorithm. It is noteworthy that the improvement is larger for the most difficult mixtures, those involving same-gender speakers.

In order to verify that the method is robust to changes in the relative gain of the signals in the mixtures, the performance was evaluated in a range of different target-to-interference ratios (TIR) (see Figure 3). The results indicate that the method works very well even when the TIR is not known a priori.

In Figure 5, the performance is measured as a function of the available training data, indicating that the method is almost converged when using 5 minutes of training data.

4. DISCUSSION

The main idea in this paper was to use sparse coding features in a linear estimation scheme. We have shown that this approach leads to better performance compared to linear regression on spectral features and compared to separation using the sparse features directly. Our results warrant further studies of the use of sparse features for speech separation, possibly using a more sophisticated estimator than the linear regression model discussed here.

The computation in the linear model is fast, since the estimation of the separation matrix is closed-form given the features. The SNMF for computing the dictionaries and the sparse feature mapping of the mixture, however, is quite expensive. A possible remedy for the latter computations could be to devise a greedy approximation.

We experimented with concatenating features across time as a simple means of modeling the temporal dynamics of speech. Doing this appears to improve performance slightly, but the effect is relatively small, confirming previous reports that the inclusion of an acoustical dynamical model yields only marginal improvements [2], [18].
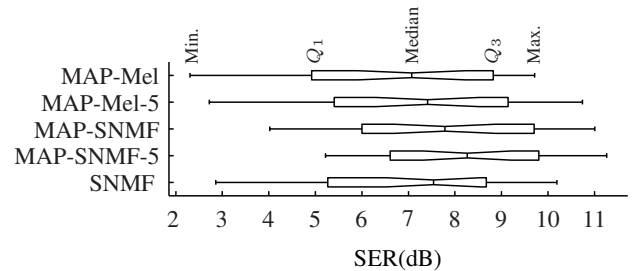

Figure 1: The distribution of the signal-to-error (SER) performance of the method for all combinations of two speakers. The mel magnitude spectrogram (MAP-Mel) and the SNMF encodings (MAP-SNMF) were used as features to the linear model. The results of using basic SNMF are given as a reference. The box plots indicate the extreme values along with the quartiles of the dB SER, averaged across sources.
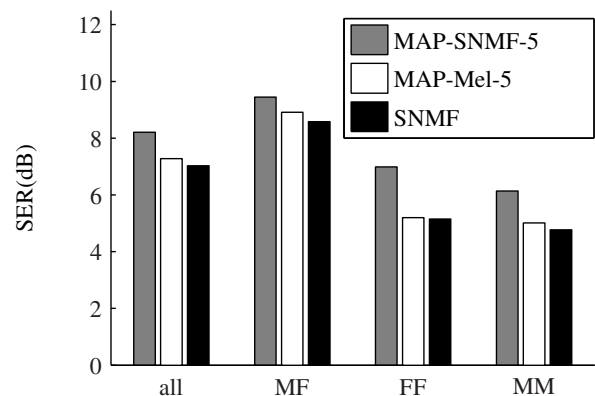

Figure 2: The performance of the methods given as signal-to-error (SER) in dB, depending on the gender of the speakers. Male and female are identified by 'M' and 'F', respectively. The improvement of MAP-SNMF-5 over MAP-Mel-5 and SNMF is largest in the most difficult (same-gender) mixtures.
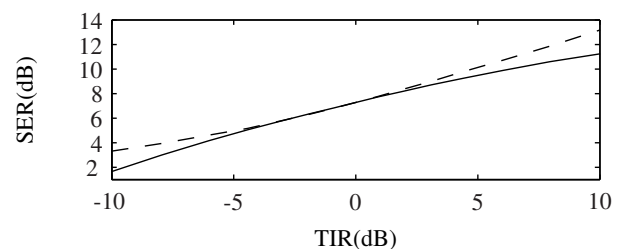

Figure 3: The performance of the MAP-Mel-5 algorithm given as the signal-to-error ratio (SER) of the target signal versus the target-to-interference ratio (TIR) of the mixture. The solid and dashed curves represent training on 0dB or the actual TIR of the test mixture, respectively. Clearly, the method is robust to a mismatch of the TIR between the training and test sets.
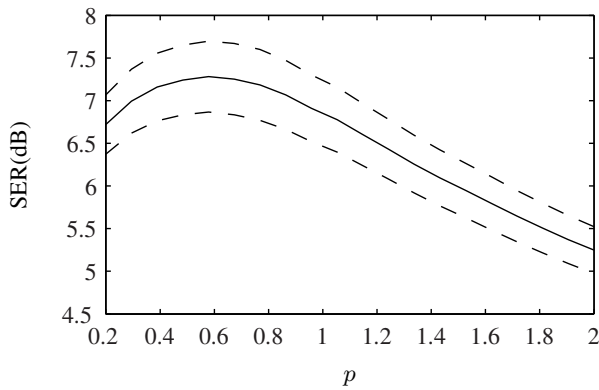
Figure 4: The effect of amplitude compression on the performance of the MAP-Mel-5 algorithm as measured in the signal-to-error ratio (SER). The optimal value of the exponent was found at $p \simeq 0.55$, in approximate accordance with Steven's power law for hearing. The dashed curve indicates the standard deviation of the mean.
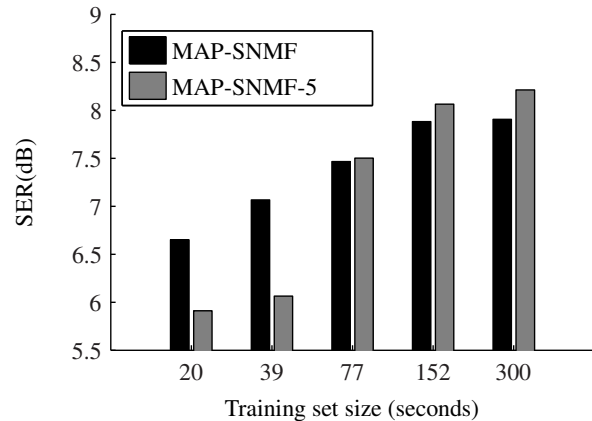

Figure 5: The learning curve of the method, measured in signal-to-error ratio (SER), as a function of the size of the training set, depending on the complexity of the method.

5. ACKNOWLEDGMENT

6. REFERENCES

[1] S. T. Roweis, "One microphone source separation," in *Advances in Neural Information Processing Systems*, 2000, pp. 793–799.

[2] T. Kristjansson, J. Hershey, P. Olsen, S. Rennie, and R. Gopinath, "Super-human multi-talker speech recognition: The IBM 2006 speech separation challenge system," in *International Conference on Spoken Language Processing (INTERSPEECH)*, 2006, pp. 97–100.

[3] G. J. Jang and T. W. Lee, "A maximum likelihood approach to single channel source separation," *Journal of Machine Learning Research*, vol. 4, pp. 1365–1392, 2003.

[4] B. A. Pearlmutter and R. K. Olsson, "Algorithmic differentiation of linear programs for single-channel source separation," in *Machine Learning and Signal Processing, IEEE International Workshop on*, 2006.

[5] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[6] P. Smaragdis, "Discovering auditory objects through non-negativity constraints," in *Statistical and Perceptual Audio Processing (SAPA)*, 2004.

[7] J. Eggert and E. Körner, "Sparse coding and NMF," in *Neural Networks, IEEE International Conference on*, vol. 4, 2004, pp. 2529–2533.

[8] T. Virtanen, "Sound source separation using sparse coding with temporal continuity objective," in *International Computer Music Conference, ICMC*, 2003.

[9] L. Benaroya, L. M. Donagh, F. Bimbot, and R. Gribonval, "Non negative sparse representation for wiener based source separation with a single sensor," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, 2003, pp. 613–616.

[10] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *International Conference on Spoken Language Processing (INTERSPEECH)*, 2006.

[11] D. Ellis, "Evaluating speech separation systems," in *Speech Separation by Humans and Machines*, P. Divenyi, Ed. Kluwer Academic Publishers, ch. 20, pp. 295–304.

[12] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[13] M. P. Cooke, J. Barker, S. P. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *submitted to JASA*.

[14] D. P. W. Ellis. (2005) PLP and RASTA (and MFCC, and inversion) in Matlab. [Online]. Available: http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat

[15] H. Hermansky, "Perceptual linear predictive (plp) analysis of speech," *The Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, 1990.

[16] D. L. Wang and G. J. Brown, "Separation of speech from interfering sounds based on oscillatory correlation," *IEEE-NN*, vol. 10, no. 3, p. 684, 1999.

[17] M. N. Schmidt and R. K. Olsson. (2007) Audio samples relevant to this paper. [Online]. Available: http://mikkelschmidt.dk/waspaa2007

[18] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Transaction on Audio, Speech and Language Processing - to appear*, 2007.