Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Balancing Gaussianity and sparseness in feature-space speaker adaptation for word prominence detection

Andrea Schnall, Martin Ernst Heckmann

2016

Preprint:

This is an accepted article published in 12. ITG Fachtagung Sprachkommunikation. The final authenticated version is available online at: https://doi.org/[DOI not available]

Balancing Gaussianity and sparseness in feature-space speaker adaptation for word prominence detection

Andrea Schnall[1]*, Martin Heckmann*[2]

[1] TU Darmstadt - Control Methods and Robotics
Holzhofallee 38, 64295 Darmstadt, Germany
[2] Honda Research Institute Europe GmbH
Carl-Legien-Str. 30, 63073 Offenbach/Main, Germany
Email: aschnall@rmr.tu-darmstadt.de, martin.heckmann@honda-ri.de

Abstract

Word prominence is an essential part of communication, e.g. to express important information. But due to speaker variations, it is difficult to extract these cues. A common method therefore is the adaptation of data from a novel speaker to a model trained from a larger pool of speakers.
For the detection via an SVM classifier we developed an adaptation method based on the radial basis function of the SVM with a Gaussian regularization, which is derived from fMLLR. With this method we could improve the word prominence detection, but, as it is a common problem, a sufficient amount of labeled data is necessary.
In this paper we show that by adding an additional regularization term, we can further improve the classification and reduce the amount of necessary data notably. Additionally, we investigate how the weighting of the different terms influences the performance and can be used to improve the results.

1 Introduction

Extracting prosodic cues like prominence in speech is a difficult task. One reason is, that every person uses slightly different features to express prominence. This can be done e.g. by raising the voice, stretching a word or even gestures. Still those prosodic cues are important in our communication since the same sentence can get a different meaning through them and integrating those cues in a speech recognition system should be beneficial. In speech processing the problem of speaker mismatch with respect to a trained model is solved by a model or feature adaptation. Established adaptation methods in speech processing are e.g. maximum likelihood linear regression (MLLR) [1], its feature-space variant, feature-space maximum likelihood linear regression (fMLLR), [1] and maximum a posteriori (MAP) [2]. However until now only a few works have employed these techniques for the adaption of prosodic features. One [3] proposes a system which adapts to a drivers voice for emotion recognition in the automotive environment. The adaptation is accomplished in the acoustic space by mean and standard deviation normalization. Another work [4] used a combination of MAP and Gaussian mixture models (GMM) for an unsupervised adaptation to a new unlabeled data set. But since that, to our knowledge, not much research has been done and there is no other work which uses speaker adaptation methods like fMLLR for the detection of prosodic cues and especially not in combination with support vector machines.
In our former experiments we recognized that for a problem case like the prominence detection, support vector machines (SVM) perform much better than a classifier like Gaussian Mixture Models, but the use of fMLLR with SVM does not lead to an improvement. Therefore we developed a method that uses a direct adaptation to the SVM boundarys [5]. To avoid overfitting we use the general fMLLR formulation as an additional term. In this paper we present an extension of our method. Since, like all adaptation methods we have to deal with the problem of estimating a high amount of parameters while having only a very small amount of speaker specific data, we added a regularization to enforce a sparse transformation matrix and thereby obtain more robust results. Additionally, we investigate how the balancing of these terms influences the performance, in particular with varying amounts of adaptation data.
The next section will present a short overview of the adaptation methods. Afterwards, we give a description of the database used, followed by a description of the audio features in section 3 and 4. Following a short description of the SVM implementation in section 5 is the presentation of the results in section 6. We will conclude with a discussion of our findings in section 7 and a subsequent conclusion in our final section.

2 Adaptation method

Speaker adaptation techniques try in general to reach speaker dependent classification performance with only a small amount of speaker specific data. As basis and reference for our method the fMLLR is used. In [5] we presented two new adaptation methods, namely the feature-space SVM adaptation and Gauss regularized feature-space SVM adaptation, that are primarily designed for SVMs with radial basis function kernels. For problems like the prominence detection, discriminative classifiers like the SVM proofed as a good choice and it is sensible to use an adaptation fitted to this classifier and also use the discriminative behavior in the adaptation.
The feature-space adaptation methods calculates a transformation matrix $\mathbf{W} = [\mathbf{A} \;\; \mathbf{b}]$, where $\mathbf{A}$ is an affine, square transformation matrix and $\mathbf{b}$ is a bias vector, to fit new data to a trained model:

$$\hat{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b} = \mathbf{W}\xi, \tag{1}$$

where $\xi = [1 \;\; \mathbf{x}^T]^T$ is the input vector extended with an extra element equal to unity.
To calculate $\mathbf{W}$, we optimize a function consisting of the main term $\mathcal{Q}_{\text{SVM}}$, which adapts the data to the radial basis function of the SVM and two regularization terms $\mathcal{Q}_{\text{G}}$ and the new proposed term $\mathcal{Q}_{\text{S}}$, weighted by λ_G and λ_S:

$$\mathcal{Q}_C = \mathcal{Q}_{\text{SVM}} + \lambda_G \mathcal{Q}_{\text{G}} + \lambda_S \mathcal{Q}_{\text{S}}. \tag{2}$$

The Optimization term of the single fSVMA, $\mathcal{Q}_{\text{SVM}}$, makes use of the radial basis function with the parameters of the trained model, the weights α_i, the class affiliation y_i, the

ISBN 978-3-8007-4275-2

© 2016 VDE VERLAG GMBH · Berlin · Offenbach

scaling factor γ_{SVM} and the support vectors $\mathbf{s}_n$. It calculates a transformation that minimizes the mismatch between the new data and the model, optimizing:

$$\mathscr{Q}_{\text{SVM}} = \sum_{t,n}^{T,N} y_t \alpha_n e^{(-\gamma_{\text{SVM}} \|\mathbf{W}\xi_t - \mathbf{s}_n\|^2)}, \tag{3}$$

with N being the number of support vectors and T the number of samples. The function is optimized using gradient descent. We iteratively adapt the rows $\mathbf{w}_i$ of $\mathbf{W}$.
The first extension term (GrfSVMA), the Gaussian regularization, uses the fMLLR to regularize the SVM adaptation in order to get a tradeoff between the fitting of the adaptation data to the SVM boundary and the more general fitting to the assumed Gaussian distribution. This has been proofed as beneficial for our high dimensional problem case [5]. The auxiliary function of the regularization term maximizes the likelihood of the adaptation data given the statistics of the old data. We maximize this auxiliary function:

$$\begin{aligned}\mathscr{Q}_{\text{G}} &= \sum_{t,m}^{T,M} \gamma_m(t)[\log \mathscr{N}(\mathbf{W}\xi_t, \mu_m, \mathbf{\Sigma}_m) + \log(\mathbf{A})] \\ &= \log|\det(\mathbf{A})| \\ &\quad - \frac{1}{2}\sum_{t,m}^{T,M} \gamma_m(t)(\mathbf{W}\xi_t - \mu_m)^T \mathbf{\Sigma}_m^{-1}(\mathbf{W}\xi_t - \mu_m),\end{aligned} \tag{4}$$

with μ and $\mathbf{\Sigma}$ being the mean and variance, i.e. the sufficient statistics when assuming a normal distribution, of the training data and the posterior probability γ of sample $\mathbf{x}_t$ being in Gaussian m. M is the number of mixtures and T the number of samples.
To estimate a model with many of free parameters and not much available data is difficult and will often tend to overfit. In classification tasks an often used technique to control the overfitting is regularization. Therefore, usually a penalty term is added to the error function in order to discourage the coefficients from reaching e.g. large values, to keep the model from getting too complex or to restrict it to bounds or smoothness. A simple example for such a penalty term is to take the sum of squares [6]. To control the influence of the regularization, compared to the original error term, a coefficient λ is used.
In our case we want to prevent a bad estimate of the high amount of parameters if not enough adaptation data is available. Other methods use therefore only a diagonal transformation matrix, but this allows only to scale the data and not to rotate it or the like. Still the diagonal is the more important part in the transformation matrix and a least square regularization would keep all values, also the diagonal, small. More preferable is a sparse matrix where especially the small off-diagonal elements are suppressed, therefor we choose the L_1-norm. Therefore, we proposed to add the following term to the cost function:

$$\mathscr{Q}_S = \sum_{i,j}^{I,J} \|w_{i,j}\|. \tag{5}$$

For the calculation of the adaptation matrix we use the row-by-row updates for adaptation matrices [7]. We use the supervised version for adaptation, taking a small labeled subset of the test data for adaptation. Thereby, we use the same amount of prominent and non-prominent samples to calculate the transformation matrix.

3 Data set

The audio-visual data was recorded as a Wizard-of-Oz experiment during a small game where the subjects moved tiles to uncover a cartoon [8]. The subjects had to give spoken advice, in a simple grammar, to a computer, such as: "place green in B one". Some of the words were misunderstood by the machine; this was verbally and visually shown. Then the subjects had to correct the sentence by repeating it and using prosodic cues to emphasize the misunderstood word as they would do with a human listener. However, correcting cue phrases such as "no I said" were not allowed. This procedure should lead to a rather natural use of prosody, creating a narrow focus condition (in contrast to the broad focus condition of the original utterance) with the corrected words marked as highly prominent.
In total 16 subjects, male and female, speaking either British or American English as native language or being either bilingual in British English/German or American English/German were recorded. Over 2500 utterances with 4 to 5 words each where used for evaluation. For current the experiments we used only the audio recordings. The speech was recorded with a distant microphone at 48 kHz and later downsampled to 16 kHz. A speech recognition system trained on the Grid Corpus [9] was used for forced alignment.
Three human annotators rated the recorded data with a 4 level scale (0-3) of prominence for each word. We calculated the inter-annotator agreement with Fleiss' kappa κ. While doing so we binarized the annotations, only differentiating between prominent and non-prominent. A word was annotated as prominent if the mean rating of all annotators was above 1.5. During the annotation we saw that on the one hand, not all speakers consistently used prominence to highlight the corrected word and, on the other hand, that for some speakers the annotators were not able to come to a consensus on the prominence of the words. Therefore, from the original 16 speakers we retained 8 which showed an agreement measured with Fleiss' κ of more than 0.55 between all annotators ($0.4 < \kappa < 0.6$ is usually considered as moderate agreement).

4 Features

For the detection of word prominence we employ the prosodic features which have been previously proposed to correlate with word prominence. The beginning and end of the word is taken from the forced alignment and used to calculate the duration of a word as well as the gap length before and after the word.
The loudness is calculated by first filtering the signal with a 12th order IIR filter following the ideas outlined in [10], followed by calculating the instantaneous energy, smoothing with a low pass filter with a cut-off frequency of 10 Hz, and finally converting the results to dB. We expected the loudness to better capture the perceptual correlates of prominence than the energy.
As described in [11], we extract the fundamental frequency f_0 (following [12]), interpolate values in the unvoiced regions via cubic splines and convert the results to semitones. To detect voicing we use an extension of the algorithm described in [13]. Another feature we use is the spectral emphasis i.e. the difference between the overall intensity and the intensity in a dynamically low-pass-filtered signal with a cut-off frequency of 1.5 f_0 [14].
From the fundamental frequency, energy and loudness, we extract functionals as described in [12] for a better prosodic

ISBN 978-3-8007-4275-2

© 2016 VDE VERLAG GMBH · Berlin · Offenbach

analysis. We extract the mean, max, min, spread (max-min) and variance along the word. Prior to the calculation of the functionals we normalize the prosodic features by their utterance mean and calculated their first and second derivative. To model the contour we calculate the DCT features from the features and retain the first 10 coefficients.
Marking the focus of a word in an utterance rendering it prominent also has an influence on the neighboring words. Modifying the articulatory parameters to raise the prominence of a segment of an utterance (hyperarticulating) is usually accompanied by a reduction of these parameters (hypoarticulation) for the neighboring segments [15, 16]. It has previously been shown that taking this context information into account is very effective for the detection of word prominence and pitch accents [17–19]. Therefore, we take not only the functionals of the current word itself for classification but also those of the previous and the following word (see [17] for details). Considering all audio features with its functionals and context features we obtain a feature vector with a dimension of 159.

5 Experimental setup

For the experiments with the SVM we used the libsvm implementation [20]. A grid search on each training data set was performed to determine the penalty parameter of the error term, C, and the variance scaling factor γ of the basis function.
In order to use the adaptation we determined mean and variance of the training data, scaled them to zero mean and unit variance and than used the same scaling factors for the test and the adaptation data. We experimented with full covariance matrices [21], but it is a well-known problem that the estimation of the covariance matrix with a small amount of adaptation data is difficult; especially, with less samples per class than dimensions [22]. So on the one hand, we did a feature reduction with a PCA transformation taking only the first 79 dimensions, because discarding the 80 higher dimensions did not change the performance of the SVM. On the other hand we assumed a diagonal covariance matrix for the training and the adaptation data.
For speaker dependent classification the data is divided in a training set, containing 75% of the data, and a test set, containing the remaining 25% of the data. A 30 fold cross validation in which the data set is always split in a way that the same number of elements is taken from both classes was run. For speaker independent classification the model is trained on 7 speakers and tested on the last (leave-one-speaker-out classification). For testing the adaptation we did a 10 fold cross validation, where we split the data of the new speaker in adaptation and testing data, taking the adaptation data to calculate the transformation matrix and the test data for evaluating the performance of the adaptation.

6 Experiments

Because the two classes, prominent and non-prominent, are highly unbalanced - approx. one times to nine - the accuracy is not a good measure to compare the results. A preference of non-prominent words would always lead to very good accuracy but would not provide an objective evaluation. Consequently, the unweighted error rate is used instead:

$$\text{unweighted error} \quad = \quad 1 - \left(\frac{tp}{tp+fn} + \frac{tn}{tn+fp} \right) * 0.5,$$

	GMM	SVM
speaker dependent	17.1%	11.5%
speaker independent	22.0%	16.4%
fMLLR (10)	22.2%	18.1%
fMLLR (40)	20.4%	16.0%

Table 1: Speaker dependent vs. speaker independent classification with SVM and GMM for prominence detection: unweighted error rate average over all speakers.

with tp: true positive, fp: false positive and tn: true negative.
As motivation for our work the results of speaker dependent and speaker independent classification are shown in Tab.1 for classification with SVM and Gaussian mixture models (GMM). The error increases from 11.5% to 16.4% for SVM and from 17.1% to 22.0% for GMM when only a speaker independent model is available. This shows that a speaker independent model is not able to capture all speaker specific prominence characteristics and the use of an adaptation method is reasonable. Also the speaker independent SVM model performs already better than the speaker dependent GMM.
Additionally the results for using fMLLR with 10 and 40 data points per class are shown. With 10 data points both errors increase, with 40 data points, we have a large performance improvement for GMM but only a small improvement for SVM. This shows that the adaption in general works but that fMLLR might not be the most suitable method im combination with SVM.
To evaluate the performance of our adaptation method we used different values for λ_G to control the Gaussian regularization and for λ_S to control the sparseness, which is shown in Tab.2. We set the value of $lamda_G$ between 0 (no Gaussian regularization) and 5 and for λ_S between 0 (no sparseness regularization) and 10.
Using our previous methods without L1-regularization, for the fSVMA ($\lambda_G = 0$, $\lambda_S = 0$) the error increased for this high-dimensional problem while for the GrfSVMA ($\lambda_S = 0$) we obtained with 40 data points a better performance, 14.8%, but not for 10. Our proposed new regularization improves the results of both methods. For fSVMA using the L_1-norm there is a smaller or, for high λ_S, no increase of the error. But for the GrfSVMA with 10 data points per class the error decreases up to to 15.1% and for 40 data points it is 14.3%.
Considering the balancing of both parameters, using 10 data points the error tends to be smaller for $\lambda_G = 3$. For 40 data points the best choice is $\lambda_G = 1$, independent of the choice of λ_S, where the best result is in combination with $\lambda_S = 5$. For $\lambda_G = 5$ the error increases compared to the other cases. For the sparseness term the error decreases with increasing λ_S until the sparseness term has too much weight, then no transformation takes place at all. This is reached for all combinations for $\lambda_S = 10$. Notable is with only 10 data points per class and $\lambda_G = 0$ this happened already for $\lambda_S = 5$ and 7. The best performance is reached for 40 data points and the combination $\lambda_G = 1$, $\lambda_S = 5$, with an error rate of 14.3%, and for 10 data points the best combination is $\lambda_G = 3$, $\lambda_S = 7$, with an error rate of 15.1%.
Generally we can say that the additional terms have more impact when a small amount of adaptation data is available. So there is a clear improvement from $\lambda_G = 1$ to $\lambda_G = 3$ for 10 data points, while it is the other way around for 40 data

ISBN 978-3-8007-4275-2

© 2016 VDE VERLAG GMBH · Berlin · Offenbach

	10 data points per class					40 data points per class				
	λ_S: 0	λ_S: 1	λ_S: 5	λ_S: 7	λ_S: 10	λ_S: 0	λ_S: 1	λ_S: 5	λ_S: 7	λ_S: 10
λ_G: 0	36,4 %	26,6 %	16,4 %	16,4 %	16.4%	27,5 %	18,6 %	20,6 %	18,3 %	16.4%
λ_G: 1	17,1 %	16,0 %	15,4 %	15,5 %	16.4%	14,8 %	14,5 %	14,3 %	14,5 %	16.4%
λ_G: 3	16,9 %	16,1 %	15,2 %	15,1 %	16.4%	15,5 %	15,2 %	14,7 %	14,6 %	16.4%
λ_G: 5	17,1 %	16,9 %	16,2 %	15,9 %	16.4%	15,5 %	15,4 %	15,1 %	15,0 %	16.4%

Table 2: Classification error for sparse GrfSVMA, using different λ_G and λ_S.

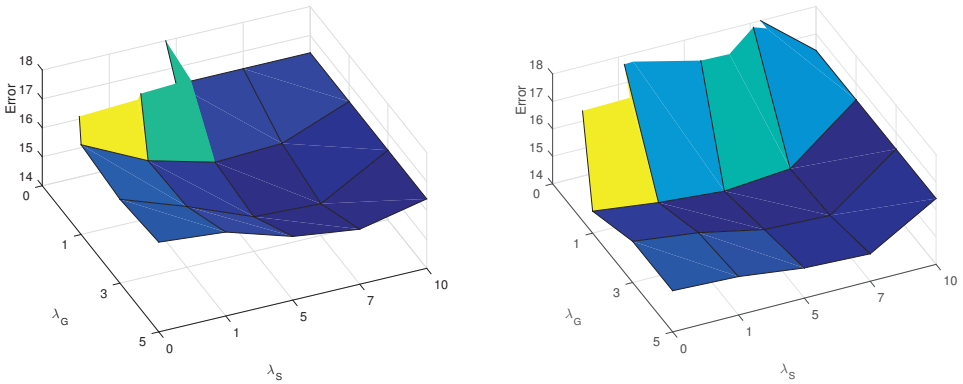


Figure 1: Classification error for sparse GrfSVMA, using different λ_G and λ_S for 10 (left) and 40 (right) data points/class.

points, which can also be seen in Fig.1.

7 Discussion

Since the difference between speaker dependent and independent classification for prominence detection is large and for this problem case discriminative classifiers performs better, our adaptation method based on SVM-RBF proved as beneficial.
The main drawback of our method has been that in such a high dimensional problem, a large amount of parameters has to be estimated from as little adaptation data as possible. For 40 data points our method GrfSVMA performed well, while for a low amount of adaptation data, e.g. 10, the adaptation failed, probably due to overfitting. Therefore we extended the method by a L_1-norm penalty term to enforce a more sparse representation of the transformation matrix and to prevent overfitting. In our experiments we can see, that this new term clearly improves results. Already with 10 data points per class using GrfSVMA the error reduces to 15.1%.
Considering the weighting of both regularization terms, our results shows that sparsity regularization is clearly necessary for these high dimensional problems. Through L1-regularization the performance notably increases for both, 10 and 40 data points, with increasing λ_S. When the weight becomes too large the performance decreases again until there is no transformation at all. It is notable that for a low amount of adaptation data and no Gaussian regularization already with a smaller weight there is no transformation. So it seems that the sparseness term has more influence when less adaptation data is available.
The additional Gaussian regularization also plays an important role for the adaptation. When using 40 data points the error decreases already without the sparseness term for $\lambda_G = 1$ and 3. Here the behavior for the different weighting parameters for the Gaussian term is different for the different amounts of adaptation data. For 10 data points the higher weighting value $\lambda_G = 3$ leads always to the best performance, while for 40 data points the performance tends to be better for $\lambda_G = 1$. So for a smaller amount of data a stronger weighting of the Gaussian term is sensible. However, we observe that for a larger amount of adaptation data setting a large value to the Gaussian term has a clear negative effect. This is in line with the observation that the FMLLR did not perform well in our case. Both observations give a clear hint that making strong assumptions that the data is Gaussian distributed is not well suited to our problem.

8 Conclusion

Our experiments show that both additional regularization terms are useful and that a balancing of both is necessary. The Gaussian regularization protects the adaptation from overfitting and already without the sparseness term improves the result when 40 data points are available. Using the L_1-norm as penalty term further improves the results for both, leading to better results, unless it is not set too high. Already a small λ_S can improve the results while at high values, e.g.$\lambda_S = 10$, there is no adaptation at all.
The combination of both terms leads to the best results: already with a small amount of adaptation data, the error decreases. When the amount of adaptation data increases a strong Gaussian regularization has a more pronounced negative effect. We will perform experiments with a wider range of available adaptation data to better investigate this behavior.

9 Acknowledgment

We thank Heiko Wersing, Lydia Fischer and Nikola Aulig for fruitful discussions. Many thanks to Merikan Koyun for the help in the data preparation.

 © 2016 VDE VERLAG GMBH · Berlin · Offenbach

References

[1] M. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer Speech and Language*, vol. 12, 1998.

[2] J. Gauvain and C. Lee, "Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.

[3] B. Schuller, "Speaker, noise, and acoustic space adaptation for emotion recognition in the automotive enviroment," *ITG-Fachtagung Sprachkommunication*, 2008.

[4] S. Ananthakrishnan and S. Narayanan, "Unsupervised adaptation of categorical prosody models for prosody labeling and speech recognition.," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 17, pp. 138–149, 2009.

[5] A. Schnall and M. Heckmann, "Speaker adaptation for word prominence detection with support vector machines," 2016. Speech Prosody, Boston.

[6] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[7] K. Sim and M. Gales, "Adaptation of precision matrix models on large vocabulary continuous speech recognition," *ICASSP*, 2005.

[8] M. Heckmann, "Audio-visual evaluation and detection of word prominence in a human-machine-interaction scenario," *INTERSPEECH, Portland, OR*, 2012.

[9] M. Cooke, J. Barker, S. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2421–2424, 2006.

[10] Replaygain, "1.0 specification."

[11] M. Heckmann, "Steps towards more natural human-machine interaction via audio-visual word prominence detection," in *2nd Workshop on Multimodal Analyses enabling Artificial Agents in Human-Machine Interaction, Singapore*, 2014.

[12] M. Heckmann, F. Joublin, and C. Goerick, "Combining rate and place information for robust pitch extraction," in *INTERSPEECH, Antwerp*, 2007.

[13] T. Kristjansson, S. Deligne, and P. Olsen, "Voicing features for robust speech detection," in *INTERSPEECH*, vol. 2, p. 3, 2005.

[14] M. Heldner, "On the reliability of overall intensity and spectral emphasis as acoustic correlates of focal accents in swedish," *Journal of Phonetics*, vol. 31, no. 1, pp. 39 – 62, 2003.

[15] Y. Xu and C. Xu, "Phonetic realization of focus in english declarative intonation," *Journal of Phonetics 33*, pp. 159–197, 2005.

[16] M. Dohen and H. Loevenbruck, "Interaction of audition and vision for the perception of prosodic contrastive focus," *Language and speech*, vol. 52, 2009.

[17] A. Schnall and M. Heckmann, "Integrating sequence information in the audio-visual detection of word prominence in a human-machine interaction scenario," in *INTERSPEECH, Singapour*, 2014.

[18] G. Levow, "Context in multi-lingual tone and pitch accent recognition," *INTERSPEECH*, pp. 1809–1812, 2005.

[19] A. Rosenberg and J. Hirschberg, "Detecting pitch accents at the word, syllable and vowel level," *HLT-NAACL*, 2009.

[20] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27;1–27;27, 2011. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[21] D. Povey and G. Saon, "Feature and model space speaker adaptation with full covariance gaussians.," in *INTERSPEECH*, ISCA, 2006.

[22] J. P. Hoffbeck and D. A. Landgrebe, "Covariance matrix estimation and classification with limited training data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, pp. 763–767, 1996.

ISBN 978-3-8007-4275-2

© 2016 VDE VERLAG GMBH · Berlin · Offenbach