Speaker Adaptation for Word Prominence Detection with Support Vector Machines

Andrea Schnall[1,2]*, 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
aschnall@rmr.tu-darmstadt.de, martin.heckmann@honda-ri.de

Abstract

In this paper we propose a new speaker adaptation method to improve the detection of prominent words in speech. Prosodic cues are difficult to extract, due to the different features different speakers are using to express for example prominence. To overcome the problem of variations from the pool of speakers used during training and those encountered during deployment, in speech recognition speaker adaptation techniques like feature-space Maximum Likelihood Linear Regression (fMLLR) turned out to be very useful. In the case of prominence detection, our former results showed that a discriminative classifier like SVM works better than GMM. Since existing adaptation methods like fMLLR are developed for GMM-HMM based classifiers and the assumption that the data has a Gaussian distribution does not hold for our data, using the fMLLR with the SVM leads not to an improvement for our problem case.
Therefore we propose a new adaptation method, which combines the fMLLR with an adaptation to the radial basis function kernel of the SVM. We investigate how this method can be used to adapt a new speaker to a speaker independent model for word prominence detection. We show that the performance improves from the speaker adaptation from 16.4% error rate to 14.4%.
Index Terms: prosody, speaker adaptation, fMLLR, SVM

1. Introduction

A very important but in speech recognition systems often neglected component in human communication is prosody. A lot of information is incorporated in this cues and could be also used for human-machine interaction systems to improve the performance and make them more intuitive. One prosodic cue we are using to determine important information is the prominence of a word. One reason that this cues are not jet often used in spoken dialog systems [1, 2] might be that it is not easy to build a general detection system since there is a large variation between different speakers.
While a speaker dependent trained classifier can reach already quite good performance results, the performance drops if a speaker independent system is used. In real life it is not useful to use speaker dependent trained models due to the high amount of labeled data which is needed to train such a model. To overcome this problem we are using the speaker independent system and adapting it with a small amount of speaker dependent data to obtain an accuracy comparable to that of a model trained on a specific speaker. Speaker adaptation is an established method in speech processing. There, often used adaptation methods are maximum likelihood linear regression (MLLR) [3] and maximum a posteriori (MAP) [4]. But until now only few work has been done to use these techniques for the adaption of prosodic features. One work [5] proposes a system which adapts to a drivers voice for emotion recognition in the automotive environment. The adaptation is done in the acoustic space by mean and standard deviation normalization. Another work [6] 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 work has been done and there is no other work which uses speaker adaptation methods like fMLLR for detection of prosody and especially not for support vector machines.
Compared to the work of [6] we investigate an adaptation that is independent of the classifier. Prominence detection is a two class problem with highly overlapping classes. Therefore our experiments showed that support vector machines (SVM) are a better choice for the classification than e.g. a Gaussian mixture model based classification. For our former experiments [7] we chose for adaptation the feature-space maximum likelihood linear regression over the in speech processing more common, model-based, methods like MLLR and MAP, since the feature space variant is in principle independent from the classifier. The fMLLR can therefore be used also in combination with SVM. Our results showed that the adaptation worked in combination with the GMM but did not improve the classification of the SVM. One reason is that the assumption of an Gaussian distribution does not fit to our data, the other might be that fMLLR does not incorporate any discriminative information. Therefore in this paper we propose a new method, based on the radial basis function parameters that is used in the trained SVM model to incorporate the information of the decision boundary in the adaptation. For better results we subsequently combine both methods.
The next section presents the database used, followed by a description of the audio features in section 3. Than we give a short overview of the adaptation methods in section 4: the fMLLR (4.1) followed by our new method (4.2) and the combined methods (4.3). Following a short description of the SVM implementation in section 5 is section 6 where the results will be presented. We will conclude with a discussion in section 7 and in a conclusion in the final section.

2. Data set

The audio-visual data was recorded as a Wizard-of-Oz experiment in 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 of the form "place green in B one". Some of the words were misunderstood by the machine, which was verbally and visually shown. Then the subjects had to correct the sentence by repeating them and using prosodic cues to emphasize the misunderstood word as they would do with a human. However correction cue phrases like "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.

For the experiments a subset of 8 subjects, male and female, speaking either British or American English as native language or being either bilingual British English/German or American English/German were recorded. Over 2500 utterances with 4 to 5 words each where used for evaluation. For the experiments in this work we used only the audio recordings. The speech was recorded with a distant microphone at 48 kHz and later down sampled to 16 kHz. A speech recognition system trained on the Grid Corpus [9] was used for forced alignment.

Three human annotators annotated 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, i.e. only differentiating between prominent and non-prominent. A word was annotated as prominent if the mean annotation 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 the annotators were for some speakers not able to come to a consensus on the prominence of the words. Therefore, from the originally 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).

3. Features

For the detection of word prominence we use 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 an 12th order IIR filter following the ideas outlined in [10], next calculating the instantaneous energy, followed by 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 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 been shown previously that taking this context information into account is very effective for the detection of word prominence and pitch accents [17, 18, 19]. Therefore, we take for classification not only the functionals of the current word itself but also those of the previous and the following word (see [17] for details).

Taking all audio features with its functionals and context features we get an feature vector with a dimension of 159.

4. Adaptation methods

In order to use the adaptation we determined mean and variance of the training data, scaled them to zero mean and unit variance and used than the same scaling factors for the test and the adaptation data.

We experimented first with full covariance matrices [20], 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 data per class than dimensions [21]. 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.

4.1. Feature-space MLLR

The first adaptation method we are using is the feature-space Maximum likelihood linear regression (fMLLR) [3]. Compared to MLLR, which transforms mean and variance of an HMM-model, it transforms directly the features and is hence independent of the classifier and can be used in combination with e.g. SVMs. To calculate the adaptation matrix we use the row-by-row updates for adaptation matrices [22]. The transformation matrix W consists of a affine matrix A and a bias vector b:

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

where $\xi = [1 \;\; x^T]^T$ is the input vector extended with an extra element equal to unity.

The transformation matrix is calculated by optimizing the auxiliary function:

$$\begin{aligned}\mathcal{Q}_{\text{ML}} &= -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_m(t)[\log\mathcal{N}(W\xi_t,\mu_m,\Sigma_m)+\log(A)]\\ &= \log|\det(A)|\\ &\quad -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_m(t)(W\xi_t-\mu_m)^T\Sigma_m^{-1}(W\xi_t-\mu_m),\end{aligned}$$

with Σ and μ being the statistics of the training data and the posterior probability γ of sample x_t being in Gaussian m. M is the number of mixtures and T the number of samples. We use the supervised version for adaptation, taking a small labeled subset of the test data for adaptation. Therefore we take the same amount of prominent and non-prominent samples to calculate the transformation matrix.

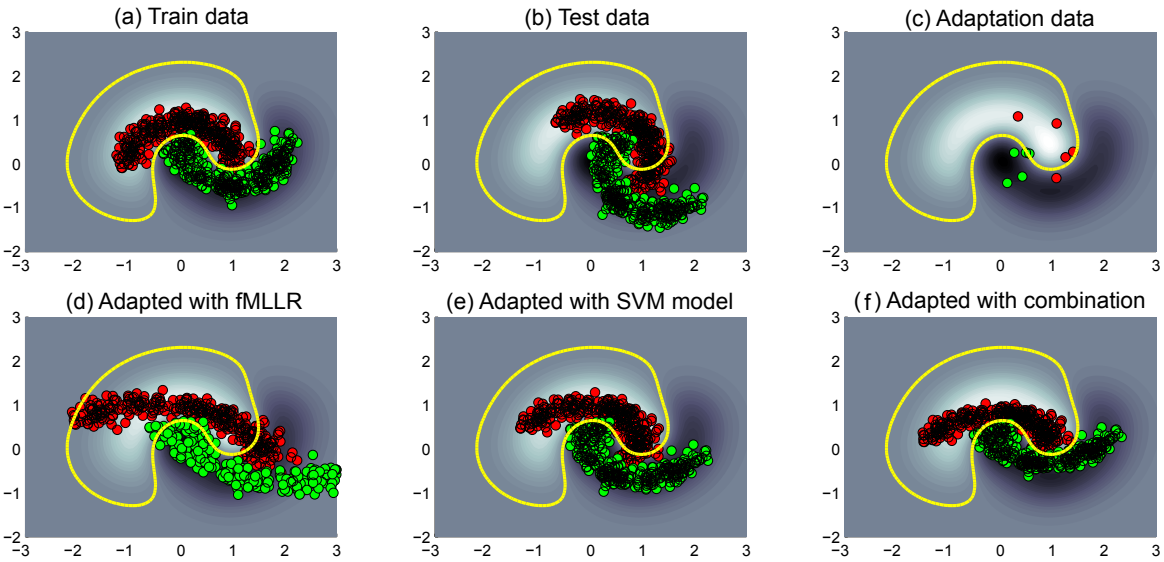

Figure 1: (a) Training data, (b) test data, (c) adaptation data, (d) data after Transformation calculated with fMLLR, (e) data after Transformation calculated based on SVM model, (f) data after Transformation calculated with the combination of both.

We also experimented with different number of Gaussian mixtures, but taking more than one per class did not improve the results.

4.2. Adaptation to SVM boundary

Since the data seems to be not well represented by a Gaussian mixture model and the fMLLR is not that suitable for a discriminative classifier, we propose to adapt the data more directly to the trained SVM model. For calculating the SVM model we use the radial basis function kernel (RBF). The RBF is defined through the support vectors s_i and the scale parameter γ_{SVM}. So the function we want to optimize to calculate the transformation matrix W is now:

$$\mathcal{Q}_{\text{SVM}} = \sum_{t}^{T} \sum_{i}^{I} y_t \alpha_i e^{(-\gamma_{\text{SVM}} \| W \xi_t - s_i \|^2)}; \qquad (2)$$

with the parameters from the learned SVM model, α_i being the weights, y_i the class affiliation and I being number of support vectors. The function is optimized by using gradient descent.

4.3. Combination of both methods

A transformation matrix that fits the adaptation data to the decision boundary will probably tend to overfit. Therefore we propose a combination of both methods, to get an tradeoff between the fitting of the adaptation data to the SVM boundary and the more general fitting to the assumed Gaussian distribution. In the new optimization function both terms are weighted by λ_1 respectively λ_2:

$$Q_C = \lambda_1 \mathcal{Q}_{\text{ML}} + \lambda_2 \mathcal{Q}_{\text{SVM}}. \qquad (3)$$

The combined term is also optimized by using gradient descent.
To illustrate how the adaptations are working we show the transformation results on the example of the two dimensional classification problem from [23] with two intertwining moon distributions belonging each two a specific class. In Fig.1 the two classes (green/red) as well as the SVM decision boundary (yellow) can be seen. (a) shows the data used for training, while (b) shows the slightly shifted and rotated (25°) data for testing. This example shows a case where an adaptation would be useful but the Gaussian assumption does not fit. Especially the estimation of the variance is difficult if only a small amount of adaptation data (c) is available. The sub images (d), (e) and (f) shows the test data after transformation with the fMLLR, the SVM based adaption and the combination of both. The corresponding classification results can be seen in Tab.1. It can be seen that the adaptation with fMLLR brings only small gain while there is a risk that the data gets to much stretched. The SVM based adaptation as well as the combination work quite well and can reach near the same performance than the training data.

(a)	(b)	amount of adaptation data per class	(c)	(d)	(e)
0.5%	15%	5	14%	1.8%	3.4%
		10	7.2%	1.5%	2.8 %

Table 1: Error for classification with the training data (a), test data (b), adapted data with fMLLR (c), SVM model (d) and combination (e).

5. Classification

As mentioned before for problems like the word prominence detection discriminative classifiers proofed as a good choice, we use a classification with support vector machines (SVM). Therefore we used the libsvm [24] with a radial basis function kernel. For the parameter C, the penalty parameter of the error term, and the variance scaling factor γ of the basis function, a grid search on the whole data set was performed.

	speaker dependent	speaker independent
GMM	17.1%	22.0%
SVM	11.5%	16.4%

Table 2: Speaker dependent vs. speaker independent classification with SVM and GMM for prominence detection: unweighted error rate average over all speakers.

6. Results

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. Instead the unweighted error rate is taken.

$$\text{precision} = \frac{tp}{tp + fp}, \quad (4)$$

$$\text{specificity} = \frac{tn}{tn + fp}, \quad (5)$$

$$\text{unweighted error} = 1 - \frac{\text{precision} + \text{specificity}}{2}, \quad (6)$$

with tp: true positive, fp: false positive and tn: true negative. For a better understanding of the possible gain of the speaker adaptation we will first show the performance difference between the classification with speaker dependent and speaker independent classification. For speaker independent classification the model is trained on 7 speakers and tested on the last (leave-one-speaker-out classification). 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, using a 30 fold cross validation in which the data set was always split such that the same number of elements is taken from both classes was run. Tab.2 shows the results for speaker dependent classification compared to the speaker independent classification with SVM and GMM. The average unweighted error rate over all speakers for speaker dependent training is 11.5% for SVM and 17.1% for GMM. If we have a speaker independent classification the result for the averaged unweighted error rate increases to 16.4% respectively 22% . Tab.3 now shows the results for the speaker independent classification for the different adaptation techniques. For the adaptation we considered 40 data points per class for the adaptation. Former experiments showed that with less adaptation data it is hard to represent this high dimensional data. The weighting parameters λ_1 and λ_2 of the combined method where, after testing different values, both set to 1 to give both parts an equal weight. Without adapting the classification error is 16.4% for SVM and 22% for GMM. Using the fMLLR did improve the results for the GMM classification from 22.0% to 20.4% but not for the SVM classification. Therefore, for the SVM, using only the adaptation to the SVM boundary the error did increase. But for the combination of the methods we got an improvement of 2% to 14.4%. Using instead the same adaptation data additional to the training data to retrain the model, there is no improvement since the amount of adaptation data is very small compared to around 10000 original data points for training.

	(a)	(b)	(c)	(d)
GMM	22.0%	20.4%	-	-
SVM	16.4%	16.3%	29.2%	14.4%

Table 3: Error for classification with the test data (a), adapted data with fMLLR (b), and SVM model (c) and combination (d).

7. Discussion

Due to the high inter speaker variations the unweighted error rate averaged over all speakers drops from 11.2% for speaker dependent to 16.2% for speaker independent training, although the amount of training data is much higher for the independent model. Therefore an adaptation should be useful. The standard fMLLR works for GMM, but since the overall results for SVM are better than GMM with adaptation, an adaptation method which is able to improve the SVM classification is desirable.
The two dimensional example with the moon shaped distribution showed, that for the case of a classification with SVM for data that is not Gaussian distributed, the standard adaptation method fMLLR is not the best choice. Our method which uses the information of the decision boundary could get a much higher performance, already with a low amount of adaptation data.
For our real data the classification is much more complex due to the high dimension and the overlapping classes. Using fMLLR with 40 data points per class to calculate the transformation matrix, did not lead to a notable improvement for SVM. A small amount of adaptation data is not able to give a good representation of the statistics of the data. Therefore we assume that the reason is that the data is not easily describe by a Gaussian model. Using only the adaptation to the SVM boundary did decline the error. The reason therefore might be that with the small amount of adaptation data, we had a strong overfitting to the adaptation data. But the combined method which on the one hand uses the information about the boundary and avoids overfitting by restricting the transformation through the constraint of the fMLLR did lead to an relative improvement of 12.5% from 16.4% to 14.4%.

8. Conclusion

In this paper we presented a new method for speaker adaptation developed for a SVM classifier with RBF kernel. To our knowledge there hasn't been any work to use common speaker adaptation methods like fMLLR for the detection of prosodic features, especially not in a classification with SVM. For problems where a differential classifier like SVM works best but the Gaussian distribution assumption does not hold, like in the case of speaker independent prominence detection, the standard method for speaker adaptation is not the best choice. We showed that our method in general works well for an example like the moon shaped distribution. In the case of the word prominence detection where there is only few adaptation data available in comparison to the feature dimensionality, the adaptation to the decision boundary leads to an overfitting. But we could show that the combination of our method with the fMLLR leads to an improvement from 16.4% to 14.4% error rate.

9. Acknowledgment

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

10. References

[1] A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, R. Bates, D. Jurafsky, P. Taylor, R. Martin, C. Van, and E. Meteer, "Dialogue act modeling for automatic tagging and recognition of conversational speech," *Computational Linguistics*, vol. 26, pp. 339–373, 2000.

[2] J. Hirschberg, D. Litman, and M. Swerts, "Characterizing and predicting corrections in spoken dialogue systems," *Comput. Linguist*, vol. 32, pp. 417–438, 2006.

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

[4] 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.

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

[6] 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.

[7] A. Schnall and M. Heckmann, "Speaker adaptation for word prominence detection," 2016, submitted to ICASSP.

[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. [Online]. Available: http://wiki.hydrogenaudio.org/

[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, 2005, p. 3.

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

[21] 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.

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

[23] Y. Ma and G. Guo, *Support Vector Machines Applications*, ser. SpringerLink : Bücher. Springer International Publishing, 2014.

[24] 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.