INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

DNN-based Feature Extraction and Classifier Combination for Child-Directed Speech, Cold and Snoring Identification

Gábor Gosztolya[1,2], *Róbert Busa-Fekete*[3], *Tamás Grósz*[1], *László Tóth*[2]

[1]University of Szeged, Institute of Informatics, Szeged, Hungary
[2]MTA-SZTE Research Group on Artificial Intelligence, Szeged, Hungary
[3]Yahoo Research, New York, NY

{ ggabor, groszt, tothl } @ inf.u-szeged.hu, busafekete@yahoo-inc.com

Abstract

In this study we deal with the three sub-challenges of the Interspeech ComParE Challenge 2017, where the goal is to identify child-directed speech, speakers having a cold, and different types of snoring sounds. For the first two sub-challenges we propose a simple, two-step feature extraction and classification scheme: first we perform frame-level classification via Deep Neural Networks (DNNs), and then we extract utterance-level features from the DNN outputs. By utilizing these features for classification, we were able to match the performance of the standard paralinguistic approach (which involves extracting thousands of features, many of them being completely irrelevant to the actual task). As for the Snoring Sub-Challenge, we divided the recordings into segments, and averaged out some frame-level features segment-wise, which were then used for utterance-level classification. When combining the predictions of the proposed approaches with those got by the standard paralinguistic approach, we managed to outperform the baseline values of the Cold and Snoring sub-challenges on the hidden test sets.

Index Terms: ComParE 2017, computational paralinguistics, Deep Neural Networks, feature extraction

1. Introduction

Traditionally, the main focus of speech technology is Automatic Speech Recognition (ASR), where the task is to create the written transcription of an audio recording (an utterance) in an automatic way. Recently, however, the extraction and identification of phenomena being present in the audio signal other than the words uttered (e.g. emotions [1], conflict intensity [2], the speaker's blood alcohol level [3] or even whether the speaker is suffering from Parkinson's [4, 5] or Alzheimer's disease [6]) has gained interest, forming the area of computational paralinguistics. The importance of this area is reflected in the fact that for several years now the Interspeech Computational Paralinguistic Challenge (ComParE) has been regularly held.

The ComParE 2017 Challenge [7] consists of three Sub-Challenges, and these tasks are highly relevant for real-life applications: in the Addressee Sub-Challenge we have to determine automatically whether the adult speaks to a child or to another adult; in the Cold Sub-Challenge speakers having a cold should be found; while in the Snoring Sub-Challenge different types of snoring have to be identified. Following the Challenge guidelines (see [7]), we will omit the description of the tasks, datasets and the method of evaluation, and focus on the techniques we applied. We should add that, unlike in a standard conference study, in this case it makes sense to experiment with several techniques at the same time, which we will indeed do.

Although both ASR and computational paralinguistics deal with recordings of human speech, they are inherently different in two different ways. Firstly, ASR focuses on the words uttered, and considers everything else (the speaker's emotional state, his cognitive load, alcohol level, etc.) as noise which is to be ignored, while in paralinguistic tasks we are interested only in the non-linguistic information present in the speech signal. The second difference is a more technical one: ASR divides the speech signal into small, equal-sized excerpts called *frames*, on which local likelihoods are estimated and combined into a variable-length, utterance-level output (the transcription). Therefore, when machine learning methods are applied in ASR, they are usually applied at the frame level. In computational paralinguistics, however, each utterance is treated as one example, from which utterance-level features have to be extracted. Classification also resembles general machine learning tasks: there are only a few hundred examples instead of millions, hence researchers prefer using Support-Vector Machines (SVMs, [8]) instead of Deep Neural Networks (DNNs, [9]).

The standard solution for utterance-level feature extraction in computational paralinguistics (see e.g. [10, 11, 12]) is to extract a huge variety of audio-based features, and then perform classification at the utterance level. Notice that no machine learning is done at the frame level; however, in ASR (and in similar tasks such as laughter detection [13, 14]) fine-tuned solutions exist on how frames should be classified. Unfortunately, these are usually ignored in computational paralinguistics, and in the notable exceptions when they are not (e.g. [15, 16]), machine learning is done in a strongly task-dependant way.

In the current study we combine the two approaches. First, following standard ASR practice, we perform frame-level classification using Deep Neural Networks. Second, based on the frame-level DNN outputs, we carry out a thresholding-based, utterance-level feature extraction step. We show that the features extracted this way can be used for utterance classification even on their own, but by combining the predictions got this way with the ones of the standard approach, the results exceed the Challenge baselines, and these scores were also achieved by fusing state-of-the-art techniques such as bag-of-audio-words [17] and end-to-end learning [18].

Unfortunately, our proposed approach cannot be applied to every task: due to the presence of overfitting, it is advisable to separate the utterances used for frame-level DNN training from those that are utilized for utterance-level classification. In the ComParE 2017 Challenge, two tasks (the **Addressee** and the **Cold Sub-Challenges**) were sufficiently large to allow such a split of the training set. For the **Snoring Sub-Challenge**, however, we propose a special feature extraction scheme which does not incorporate frame-level machine learning.

http://dx.doi.org/10.21437/Interspeech.2017-905

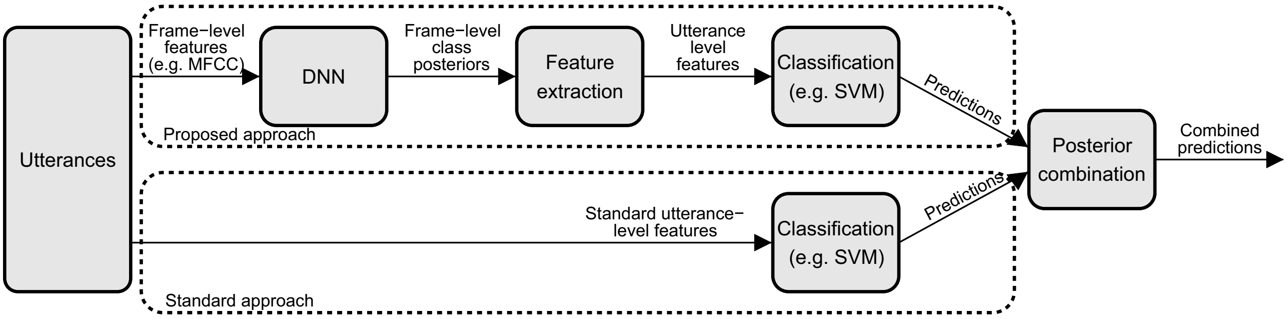

Figure 1: *The workflow of the proposed paralinguistic processing scheme.*

2. DNN-based Feature Extraction Scheme

Next, we describe the proposed feature extraction and classification approach. (For the general scheme of the proposed workflow, see Fig. 1.) For the first step, we train a Deep Neural Network at the frame level. Then, for the second step, we extract a number of (utterance-level) features from the DNN outputs, which are used to train a Support-Vector Machine to predict the actual paralinguistic phenomenon. Lastly, we combine the predictions with those obtained using standard utterance-level features.

2.1. Frame-level Classification

For the first step of our proposed workflow we train a Deep Neural Network with standard frame-level features (e.g. MFCC, PLPand FBANK [19]) as input, while the output neurons correspond to the actual, utterance-level class label for each frame. Of course, we cannot expect that the frames will be classified very accurately: in most paralinguistic tasks, the actual phenomenon we are looking for (e.g. breath intakes for physical load, cough events for having a cold, and hesitation in dementia) is not present in every part of the utterance. Still, since DNNs have proven to be quite robust, we may confidently expect them to find some locations within the utterance that are specific to the given class, and this will be reflected in the frame-level DNN outputs.

2.2. Thresholding-Based Utterance-Level Feature Extraction and Classification

For the next step we create features from the frame-level DNN outputs, and these features will be used for *utterance-level* classification. The most straightforward solution is to classify each frame based on the DNN likelihood scores, and count the ratio of frames classified as each possible class, or to aggregate the frame-level likelihoods via arithmetic or geometrical mean. However, the posterior estimates provided by a DNN contain valuable information, and this information should not be discarded. Due to this, we propose using *several* different threshold values. That is, using the step size parameter s, first we count the number of frames where the DNN output corresponding to the first class is greater than or equal to s, we divide it by the total number of frames in the utterance, and then use this value as the first newly extracted feature. Next, we repeat this step using the values $2 \cdot s, 3 \cdot s, \ldots, 1$ as thresholds and using all the classes. Doing this for all the utterances, we extract a new feature set for all the utterances, which can be used for performing classification for the third step by some machine learning method such as DNNs or SVMs.

2.3. Classifier Output Combination

Although using the frame-level DNN outputs may prove to be beneficial for classification, we should not discard all other kinds of features: optimality is probably achieved via a combination of the two approaches. One possible way of combining them is to merge the *feature vectors* of each utterance, and train one classifier model. However, often it is more beneficial training separate machine learning models for different types of features, as these may require different meta-parameter settings for optimal performance. Therefore we suggest training one machine learning method using the standard utterance-level features like that proposed in the ComParE baseline studies [11, 12, 7], and train a separate one using the features extracted as described in Section 2.2. To combine the outputs of the two models, we suggest taking the weighted mean of the (utterance-level) posterior scores of the two models, which is a simple-yet-robust technique (see e.g. [20]).

3. Experiments (Addressee and Cold)

3.1. DNN Parameters

At the frame level we trained a Deep Neural Network with 3 hidden layers, each containing 256 rectified neurons [21], while we applied the softmax function in the output layer. DNN training was done with our custom implementation for Nvidia GPUs, which achieved good accuracy scores on several tasks and datasets (e.g. [22, 23]). We used the standard 39-sized MFCC + Δ + $\Delta\Delta$ feature set;training was done with a 15-frames wide sliding window. Note that we did not fine-tune the DNN training meta-parameters, since we were interested only in the *trend* of the frame-level DNN outputs. To avoid overfitting in the subsequent classification process, we just used a subset of the training set for training these DNNs: 1000 randomly selected recordings were used for the Addressee Sub-Challenge, while we used 2000 recordings for the Cold Sub-Challenge. Of course, it would have been best to split the training set based on speakers, but speaker information was not available for either corpora.

3.2. Feature Extraction and Classification

For the next feature extraction step, we used a step size of 0.02 to threshold the DNN outputs, resulting in 50 features for each class; since for both the Addressee and the Cold Sub-Challenge there were only two classes, it was enough if we performed the feature extraction step for one class only. After standardization (i.e. transforming the feature vectors so as to have a zero mean and unit variance), we trained a Support-Vector Machine, using

Table 1: *The results obtained on the Addressee Sub-Challenge*

Approach	Dev.	Test
ComParE feature set	59.1%	—
Thresholded feature set	56.0%	—
Frame-level DNN outputs (mean)	53.8%	—
Frame-level DNN outputs (product)	53.8%	—
Frame-level DNN outputs (majority)	53.2%	—
ComParE feature set (downsampled)	61.7%	—
Thresholded feature set (downsampled)	58.8%	—
ComParE + DNN-based (combined)	62.0%	67.7%
ComParE baseline	66.4%	70.2%

Table 2: *The results obtained on the Cold Sub-Challenge*

Approach	Dev.	Test
ComParE feature set	58.3%	—
Thresholded feature set	61.1%	—
Frame-level DNN outputs (mean)	52.9%	—
Frame-level DNN outputs (product)	52.6%	—
Frame-level DNN outputs (majority)	53.1%	—
ComParE feature set (downsampled)	64.0%	—
Thresholded feature set (downsampled)	65.0%	—
ComParE + DNN-based (combined)	65.8%	72.0%
ComParE baseline	65.2%	71.0%

the LibSVM [24] library. We used the nu-SVM method with a linear kernel; the value of C was tested in the range $10^{\{-5,\ldots,1\}}$, just like that in our previous paralinguistic studies (e.g. [20, 25, 26]).

Note that there was inevitably some overfitting present in the frame-level DNN training process; this way, the DNN-based thresholding feature extraction step returned biased feature values for the utterances which were used for DNN training. Therefore we excluded these utterances from the utterance-level classification step, leaving 2742 and 7505 utterances for training, the Addressee Sub-Challenge and Cold Sub-Challenge, respectively. We chose the C value which gave the highest accuracy score on the development set; then we trained SVM models using the examples of both the training and development sets, and this model was used for making predictions for the test set.

As a reference, we also tried combining the DNN outputs in various ways: we took their mean for each class within the given utterance, we combined them by multiplication (which is the standard way of frame-level posterior aggregation), and we experimented with choosing the most probable class for each frame and then using simple majority voting of the frame-level class label hypotheses.

We also tested a combination of the results got by the proposed feature extraction scheme and those got with the standard, 6373-long feature set. For each example, we took the weighted mean of the posterior scores got by the two approaches; then the optimal weights were determined on the development set.

3.3. Instance Sampling

Another special aspect of these datasets was that the distribution of the two classes was quite imbalanced, which might reduce the performance of the classifier method used. Following our previous experiments (e.g. [26]), we decided to opt for *downsampling*: for SVM model training, we used all the training examples of the rarer classes (i.e. "Adult-Directed Speech" for the Addressee Sub-Challenge and "Cold" for the Cold Sub-Challenge), and discarded examples from the more frequent classes to exactly balance the training data. Since this sampling scheme introduced a further random factor to the training process, while also reducing the variability of the training samples, we repeated this process 100 times, and averaged out the resulting posterior scores of all the models.

3.4. Results

Tables 1 and 2 show the results obtained in the Addressee and in the Cold Sub-Challenges, respectively. The results for the two datasets have a lot in common: firstly, it is clear that only using the frame-level DNN outputs and combining them by averaging out the posteriors or taking their product yielded far worse results than what could be obtained by using the baseline, 6373-long ("ComParE") feature set. Since both tasks are binary classification ones, the UAR scores in the range $52.6 - 53.8\%$ reflect a pretty low performance (slightly above chance level). However, when we extracted the new utterance-level feature set from the frame-level DNN outputs, the resulting UAR values were much better; in the case of the Cold Sub-Challenge, we could even outperform the one got by the ComParE feature set.

The performance of the two SVM-based methods were further increased by downsampling, which is easy to understand for the Cold Sub-Challenge, where only 10% of the examples belonged to the "Cold" class; but even in the Addressee Sub-Challenge, training several models using equal class distribution and averaging out their posterior values brought a 2.5% improvement. Combining the two models trained on the two feature sets brought a further increase in the UAR values; in the case of the Cold Sub-Challenge, we managed to increase the scores on the test set as well, achieving 72.0% with out first submission. We would like to emphasize that the baseline 71.0% score is already a fusion of three approaches: besides using the model trained on the standard ComParE feature set, the predictions got via end-to-end learning and bag-of-audio-words representation were also fused into a final prediction vector. Our approach, however, performed significantly better.

The results are more difficult to interpret in the case of the Addressee Sub-Challenge. It is more interesting since the two tasks were quite similar: both were binary classification ones with a slightly (Addressee) or heavily (Cold) unbalanced class distribution, and having plenty of training examples available. In our opinion, this low performance reveals one of the weak points of our approach: the overfitting of frame-level DNNs.

Recall that, after training the frame-level DNNs, we evaluate them for all instances, extract a new feature set from the DNN outputs, and train a new classifier model for utterance classification. Since DNNs are prone to overfitting, the utterances used for DNN training cannot be used while training the second model, as the DNN outputs for their frames are biased towards the correct class. To avoid this, we used only 1000 and 2000 utterances for DNN training, the Addressee and Cold Sub-Challenges, respectively, which were then excluded during the training of the utterance-level classifiers.

DNNs, however, can overfit to other phenomena as well, such as speakers; that is, they tend to perform better for speakers present in the training set, even if the given utterance of the speaker was not used during training. The audio files used in the Addressee Sub-Challenge were from only 61 homes, and although each speaker occurred only in the training, development or test set, we could not split the training set further into

Table 3: *The results obtained with DNNs*

Sub-Challenge	Approach	Dev.	Test
Addressee	DNN + prob. sampl.	61.3%	68.1%
	ComParE (6373)	61.8%	67.6%
	ComParE baseline	66.4%	70.2%
Cold	DNN + prob. sampl.	68.1%	64.3%
	ComParE (6373)	64.0%	70.2%
	ComParE baseline	65.2%	71.0%

speaker-independent subsets, since speaker information was not provided. This overfitting can be observed in the DNN outputs: combining the frame-level posteriors for utterance classification yielded UAR scores of $68.6 - 69.6\%$ for the 2742 utterances of the training set (i.e. utterances not used for DNN training), while for the development set these values fell in the range $53.2 - 53.8\%$. DNN overfitting was not a problem for the Cold Sub-Challenge, though, due to the large numbers of speakers (630) being present there.

3.5. Experiments with Deep Neural Networks

For these two Sub-Challenges, we also experimented with the use of Deep Neural Networks. We followed the approach of DNN training that proved to be quite successful in the past ComParE Challenges [26, 27]; namely, we trained 10 neural networks for each task, having three hidden layers and each hidden layer consisting of 1000 rectified neurons. We utilized the standard 6373-sized feature set provided by the organizers, and to balance class distribution, we applied the probabilistic sampling technique [28, 29] during training with $\lambda = 1$. Examining the results (see Table 3) we can see that this approach yields results similar to those of SVM with this feature set (case *ComParE (6373)*), but lags behind the official Challenge baseline, which combines different paralinguistic techniques.

4. Experiments (Snoring)

The Addressee and Cold Sub-Challenges differed from the typical paralinguistic tasks in that there was a huge amount of utterances available, allowing large training, development and test sets. In the Snoring Sub-Challenge, however, we had 828 recordings overall, which along with the small average duration of the recordings made training a frame-level DNN unfeasible.

In our contribution to this sub-challenge, we implemented another idea. Since each recording contained only one snore event, we assumed that snore events of the same type (i.e. belonging to the same class) display similar patterns over time. To exploit this, for the first step we calculated frame-level features for all the utterances. We chose the feature set proposed by Schuller et al. [30] in the Vocalization Sub-Challenge of ComParE 2013, because we found it a quite exhaustive and robust one in our previous studies (e.g. [31, 32]). It consisted of the frame-wise 39-long $MFCC + \Delta + \Delta\Delta$ feature vector along with voicing probability, HNR, F0 and zero-crossing rate, and their derivatives. To these 47 features their mean and standard derivative in a 9-frame long neighbourhood were added, resulting in a total of 141 features. We extracted this feature set with the tool OpenSMILE [33].

For the second step we divided each utterance into 10 equal-sized parts, and simply averaged out each feature in each window; by extending this utterance-level feature set with the length (number of frames) of the utterance, we ended up with 1411 attributes overall. Using this feature set, we trained an SVM model, which could be used for making predictions both on the development set and on the test set. Note that, although there was a huge class imbalance in this task as well, we considered downsampling pretty useless, considering the low number of training examples belonging to the **Tongue** class.

Table 4: *The results obtained on the Snoring Sub-Challenge*

Approach	Dev.	Test
ComParE feature set	41.4%	—
Frame-based feature set	48.3%	—
ComParE + Frame-based (combined)	49.3%	64.0%
ComParE baseline	40.6%	58.5%

Like for the other two sub-challenges, next we trained an SVM model on the standard, 6373-long feature set; the combination of the two predictions was again done by taking the weighted mean of their posteriors, for which the optimal weights were determined on the development set.

4.1. Results

Table 4 shows the results obtained on the Snoring Sub-Challenge. It is clear that the UAR values achieved on both sets are much better than the baseline ones. As usual, the improvement is higher on the development set, on which we tuned our meta-parameters (e.g. number of segments or complexity of SVM), but our results are higher on the test set as well: our score of 64.0% is much higher than the baseline value of 58.5%, meaning a 13% improvement.

5. Conclusions

In the area of computational paralinguistics, besides following the standard approach of extracting a huge variety of standard utterance-level features, usually other task-specific steps are required to achieve state-of-the-art performance. In this study we proposed a two-step feature extraction scheme, where first we perform frame-level classification by DNNs. Then, for the second step we extract several utterance-level features from the frame-level DNN outputs, used for (utterance-level) classification. By following this approach in the Cold Sub-Challenge of the Interspeech Computational Paralinguistic Challenge 2017, we achieved a significant improvement over the baseline value on the test set. However, in the technically quite similar Addressee Sub-Challenge we were unable to even match the baseline score on the test set, which is, in our opinion, due to the low number of speakers being present in the training set, and the tendency of overfitting of frame-level DNNs. In the last sub-challenge (Snoring), due to the low number of recordings, we could not apply this approach. We extracted a number of frame-level features instead, which we averaged out in specific segments of each snore recording. By training a separate SVM model on this newly extracted feature set and combining its predictions with the SVM model trained in the standard way, we got 13% in terms of relative error reduction on the test set.

6. Acknowledgements

The Titan X graphics card used for this study was donated by the NVIDIA Corporation. Tamás Grósz was supported by the ÚNKP-16-3 New National Excellence Programme of the Ministry of Human Capacities.

7. References

[1] S. Tóth, D. Sztahó, and K. Vicsi, "Speech emotion perception by human and machine," in *Proceedings of COST Action*, Patras, Greece, 2012, pp. 213–224.

[2] H. Kaya, T. Özkaptan, A. A. Salah, and F. Gürgen, "Random discriminative projection based feature selection with application to conflict recognition," *IEEE Signal Processing Letters*, vol. 22, no. 6, pp. 671–675, 2015.

[3] D. Bone, M. Li, M. P. Black, and S. S. Narayanan, "Intoxicated speech detection: A fusion framework with speaker-normalized hierarchical functionals and GMM supervectors," *Computer, Speech & Language*, vol. 28, no. 2, pp. 375–391, 2014.

[4] J.-R. Orozco-Arroyave, J. Arias-Londono, J. Vargas-Bonilla, and E. Nöth, "Analysis of speech from people with parkinsons disease through nonlinear dynamics," in *Proceedings of NoLISP*, 2013, pp. 112–119.

[5] D. Sztahó, G. Kiss, and K. Vicsi, "Estimating the severity of Parkinson's disease from speech using linear regression and database partitioning," in *Proceedings of Interspeech*, Dresden, Germany, 2015, pp. 498–502.

[6] I. Hoffmann, D. Németh, C. Dye, M. Pákáski, T. Irinyi, and J. Kálmán, "Temporal parameters of spontaneous speech in Alzheimer's disease," *International Journal of Speech-Language Pathology*, vol. 12, no. 1, pp. 29–34, 2010.

[7] B. Schuller, S. Steidl, A. Batliner, S. Hantke, E. Bergelson, J. Krajewski, C. Janott, A. Amatuni, M. Casillas, A. Seidl, M. Soderstrom, A. S. Warlaumont, G. Hidalgo, S. Schnieder, C. Heiser, W. Hohenhorst, M. Herzog, M. Schmitt, K. Qian, Y. Zhang, G. Trigeorgis, P. Tzirakis, and S. Zafeiriou, "The INTERSPEECH 2017 computational paralinguistics challenge: Addressee, Cold & Snoring," in *Proceedings of Interspeech*, 2017, pp. 1–5.

[8] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[9] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[10] B. Schuller, S. Steidl, A. Batliner, J. Epps, F. Eyben, F. Ringeval, E. Marchi, and Y. Zhang, "The INTERSPEECH 2014 computational paralinguistics challenge: Cognitive & physical load," in *Proceedings of Interspeech*, 2014, pp. 427–431.

[11] B. Schuller, S. Steidl, A. Batliner, S. Hantke, F. Hönig, J. R. Orozco-Arroyave, E. Nöth, Y. Zhang, and F. Weninger, "The INTERSPEECH 2015 computational paralinguistics challenge: Nativeness, Parkinson's & eating condition," in *Proceedings of Interspeech*, 2015, pp. 478–482.

[12] B. Schuller, S. Steidl, A. Batliner, J. Hirschberg, J. K. Burgoon, A. Baird, A. Elkins, Y. Zhang, E. Coutinho, and K. Evanini, "The Interspeech 2016 computational paralinguistics challenge: Deception, sincerity & native language," in *Proceedings of Interspeech*, San Francisco, CA, USA, 2016, pp. 2001–2005.

[13] H. Salamin, A. Polychroniou, and A. Vinciarelli, "Automatic detection of laughter and fillers in spontaneous mobile phone conversations," in *Proceedings of SMC*, 2013, pp. 4282–4287.

[14] G. Gosztolya, A. Beke, T. Neuberger, and L. Tóth, "Laughter classification using Deep Rectifier Neural Networks with a minimal feature subset," *Archives of Acoustics*, vol. 41, no. 4, pp. 669–682, 2016.

[15] H. Kaya, A. A. Karpov, and A. A. Salah, "Fisher Vectors with cascaded normalization for paralinguistic analysis," in *Proceedings of Interspeech*, 2015, pp. 909–913.

[16] M.-J. Caraty and C. Montacié, *Detecting Speech Interruptions for Automatic Conflict Detection*. Springer International Publishing, 2015, ch. 18, pp. 377–401.

[17] M. Schmitt, C. Janott, V. Pandit, K. Qian, C. Heiser, W. Hemmert, and B. Schuller, "A bag-of-audio-words approach for snore sounds' excitation localisation," in *Proceedings of ITG Speech Communication*, Paderborn, Germany, 2016, pp. 230–234.

[18] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Proceedings of ICASSP*, Shanghai, China, 2016.

[19] S. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book*. Cambridge, UK: Cambridge University Engineering Department, 2006.

[20] G. Gosztolya, T. Grósz, Gy. Szaszák, and L. Tóth, "Estimating the sincerity of apologies in speech by DNN rank learning and prosodic analysis," in *Proceedings of Interspeech*, San Francisco, CA, USA, Sep 2016, pp. 2026–2030.

[21] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier networks," in *Proceedings of AISTATS*, 2011, pp. 315–323.

[22] L. Tóth, "Phone recognition with hierarchical Convolutional Deep Maxout Networks," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2015, no. 25, pp. 1–13, 2015.

[23] T. Grósz, R. Busa-Fekete, G. Gosztolya, and L. Tóth, "Assessing the degree of nativeness and Parkinson's condition using Gaussian Processes and Deep Rectifier Neural Networks," in *Proceedings of Interspeech*, Dresden, Germany, Sep 2015, pp. 1339–1343.

[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. 1–27, 2011.

[25] G. Gosztolya, "Conflict intensity estimation from speech using greedy forward-backward feature selection," in *Proceedings of Interspeech*, Dresden, Germany, Sep 2015, pp. 1339–1343.

[26] G. Gosztolya, T. Grósz, R. Busa-Fekete, and L. Tóth, "Determining native language and deception using phonetic features and classifier combination," in *Proceedings of Interspeech*, San Francisco, CA, USA, Sep 2016, pp. 2418–2422.

[27] ——, "Detecting the intensity of cognitive and physical load using AdaBoost and Deep Rectifier Neural Networks," in *Proceedings of Interspeech*, Singapore, Sep 2014, pp. 452–456.

[28] S. Lawrence, I. Burns, A. Back, A. Tsoi, and C. Giles, "Chapter 14: Neural network classification and prior class probabilities," in *Neural Networks: Tricks of the Trade*. Springer, 1998, pp. 299–313.

[29] L. Tóth and A. Kocsor, "Training HMM/ANN hybrid speech recognizers by probabilistic sampling," in *Proceedings of ICANN*, 2005, pp. 597–603.

[30] B. Schuller, S. Steidl, A. Batliner, A. Vinciarelli, K. Scherer, F. Ringeval, M. Chetouani, F. Weninger, F. Eyben, E. Marchi, H. Salamin, A. Polychroniou, F. Valente, and S. Kim, "The Interspeech 2013 Computational Paralinguistics Challenge: Social signals, Conflict, Emotion, Autism," in *Proceedings of Interspeech*, 2013.

[31] G. Gosztolya, R. Busa-Fekete, and L. Tóth, "Detecting autism, emotions and social signals using adaboost," in *Proceedings of Interspeech*, Lyon, France, Aug 2013, pp. 220–224.

[32] G. Gosztolya, "On evaluation metrics for social signal detection," in *Proceedings of Interspeech*, Dresden, Germany, Sep 2015, pp. 2504–2508.

[33] F. Eyben, M. Wöllmer, and B. Schuller, "Opensmile: The Munich versatile and fast open-source audio feature extractor," in *Proceedings of ACM Multimedia*, 2010, pp. 1459–1462.