DOI: 10.17185/duepublico/42385

Comparison of Different Modeling Techniques for Robust Prototype Matching of Speech Pitch-Contours

Alicia Lotz [*] **Ingo Siegert** [*] **Andreas Wendemuth** [*]

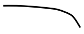
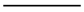
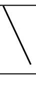
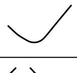
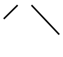
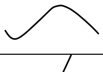
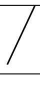
[*] *Cognitive Systems Group, Institute of Information and Communication Engineering, Otto-von-Guericke University, 39016, Magdeburg, Germany (e-mail: <firstname>.<lastname>@ovgu.de).*

Abstract: In verbal interactions between humans (HHI), and between humans and computers (HCI), a multitude of information is being transmitted. Speech conveys, besides the pure textual information, additional details regarding the speaker's feelings, believes, and social relations. Intonation reveals functional details about the speakers' communicative relation and their attitude towards the ongoing dialogue, such as affirmation, disagreement or the wish of turn-taking. Since the intonation of full words is influenced by semantic and grammatical information, it is advisable to rather investigate the intonation and corresponding functional meaning of so-called discourse particles (DPs) such as "hm" or "uhm". They cannot be inflected but can be emphasized, and the interlocutors are able to differentiate the functional meanings of DPs solely from their intonation. To take advantage of this relation in automatic dialogue processing, the goal of this investigation is to enable an automatic classification of the functional meaning of the DP "hm" from its intonation. The acoustic intonational curve can be represented using the pitch-values extracted from the raw speech material. Three different classification methods will be presented and compared to evaluate the best one. Furthermore, to ensure the reliability of the classifier both for HHI and HCI, and to gain more training data, cross-validating tests were carried out on two publicly available datasets containing HCI and HHI.

1. INTRODUCTION

Human-computer interaction recently received increased attention and the dissemination of technical assistance systems is increasingly growing in recent years. It ranges from use in household appliances, smart phones and medical systems and is under research for social robots and production support (cf. Bächler et al. [2015], Benninghoff et al. [2013]). A recent research direction is to make the operation of technical systems as simple as possible and ultimately enable a natural (human-like) interaction (cf. Skantze et al. [2015], Biundo and Wendemuth [2015]). Therefore, speech is seen as a very important information channel, as it does not only transmit the actual content.It is known from speech based human-human interaction (HHI) that semantic and prosodic cues effectively communicate certain dialog functions such as attention, understanding or attitudinal reactions of a speaker (cf. Allwood et al. [1992]). Furthermore, it is assumed that short feedback signals are uttered in situations of a higher cognitive load where a more articulated answer cannot be given (cf. Corley and Stewart [2008]). By considering the occurrence and meaning of these cues, the communication is facilitated. This could help future "cognitive" systems to better understand the human (cf. Baranyi et al. [2015]). Among the earlier mentioned semantic and prosodic cues the discourse particle (DP), in particular the feedback signal "hm", is investigated in detail (cf. Benus et al. [2007], Kehrein and Rabanus [2001]). These DPs cover specific dialogical functions and occur at crucial communicative points (cf. Ladd [1996], Schmidt [2001]). Additionally, the intonation of this type of particles is largely free of lexical and grammatical influences and can be seen as "pure intonation" (Schmidt [2001]).

Tab. 1. Form-function relation of DP "hm" according to Schmidt [2001], the terms are translated into appropriate English ones.

Name	idealized pitch-contour	Description
DP1		attention
DP2		thinking
DP3		finalization signal
DP4		confirmation
DP5		decline
DP6		positive assessment
DP7		request to respond

For the German language Schmidt empirically discovered seven types of form-function relations on the isolated DP "hm" (cf. Table 1, Schmidt [2001]). Several consecutive studies confirmed this form-function relation in human-

human interaction (HHI). One investigation presented by Kehrein and Rabanus examined this relation on four different conversational styles: talk-show, interview, theme-related talk, and informal discussion, from various German sources. They confirmed the form-function relation on manual extracted and labeled particles (cf. Kehrein and Rabanus [2001]). Another investigation, carried out by Paschen, showed that the frequency of the different dialogical functions is depending on the conversation type (cf. Paschen [1995]). By examining nearly 3 000 samples of "hm"s in different German conversations, the author could show a relation of conversation style and the use of different functionals: confirmation signs dominate in conversations of narrative or cooperative character whereas in argumentative ones turn holding signals are more frequent.

In contrast to HHI, where the interaction partner is able to understand the DPs and is using them himself, in human-computer interaction (HCI) the technical system as dialog partner is mostly neither able to understand these cues nor using them. But, nevertheless DPs are verifiably used in both types of interaction HHI and human-computer interaction (HCI) (cf. Fischer [2000], Siegert et al. [2013], Siegert et al. [2014b]). One of the very first studies dealing with the occurrence of DPs during a HCI investigated the functional meaning of DPs and concluded that the number of partner-oriented signals decreases while the number of signals indicating a talk-organizing, task-oriented, or expressive function is increasing (cf. Fischer et al. [1996], Fischer [2000]). The other two studies presented show a connection between the occurrence of DPs and the cognitive load of the human communication partner (cf. Siegert et al. [2014b]) as well as first attempts to automatically distinguish the most frequent form-function relation connected to cognitive load (DP-2, thinking) from all other meanings (cf. Siegert et al. [2013]). As the studies presented advise that the DPs are also occurring frequently in HCI while different functional meanings are used as well, the DPs are very helpful in assessing the HCI. In contrast to the previous studies, the present one discusses different approaches to develop a classifier for the automatic recognition of all occurring form-function relations in HCI.

The remainder of the paper is structured as follows: Section 2 shortly describes the utilized datasets providing the train and test samples of the DPs. In Section 3, a brief description of the classifier and the used pitch-extraction and pre-processing methods is stated. Afterwards, the three different methods for a robust form-function evaluation are presented in section 4. In section 5, the manual annotation conducted to generate the ground truth for our recognition experiments is presented. Section 6 presents the correspondence values between the ground truth and the three different form-function evaluation methods. Furthermore, the results are compared and discussed. Finally, section 7 concludes the paper and provides an outlook for further research directions.

2. DATASETS

The analyses presented in this paper are conducted on two datasets. The first dataset covers interactions between humans and a computer, the second one covers interactions between two humans specifically designed for the study of feedback signals.

The first dataset utilized is the *LAST MINUTE Corpus (LMC)* (cf. Rösner et al. [2015]). It contains multimodal recordings of 130 German speaking subjects in a so called Wizard-of-Oz experiment. The setup revolves around an imaginary journey to the unknown place "Waiuku", which the subjects have won. Each experiment takes about 30min. With the help of an adaptable technical system, the subjects have to prepare the journey, by packing the suitcase, and select clothing and other equipment, using voice commands. This dataset was selected, as it represents a naturalistic HCI (cf. Siegert et al. [2012]) and is already the object of examination regarding affective state recognition (cf. Frommer et al. [2012], Siegert et al. [2014a]) and linguistic turns (cf. Rösner et al. [2012]). For a sub-set of the corpus a total number of 259 DPs of the type "hm" are extracted from 25 hours of the HCI speech material received from 56 subjects.

The second dataset is the *ALICO Corpus*. It is recorded to investigate feedback behavior changes in HHI under the condition that the listener is distracted (cf. Buschmeier et al. [2014]). The corpus comprises 2×25 German dialogues. For all dialogues one person is telling a story while the interlocutor is listening. For each couple of speaker and listener two dialogues are recorded. In the first one the listener gets the instruction to pay attention, make remarks and ask questions. For the second one an additional distractive task is given to the listener. He is instructed to press a button on a hidden counter every time the story teller utters a word starting with the letter 's'. A sub-set of 40 dialogues is annotated, resulting in 1 505 feedback signals from 3 hours of speech material (cf. Buschmeier et al. [2014]). Out of these signals 537 are marked as "hm" and used for the investigation presented in this paper.

3. THE DP-CLASSIFIER

In this section the basic implementation of the classifier as stated in Lotz [2014] and Lotz et al. [2016] is presented. The algorithm is structured in three stages as pictured in Fig.1. First, the pitch-values are estimated. Second, to process errors and reduce the complexity of the function analysis a pre-processing of the pitch-contours is conducted. Finally, the function analysis algorithm assigns one of the seven form-function-prototypes discovered by Schmidt to the present pitch-contour (cf. Table 1). To verify the results the output of the classifier is verified against the ground truth gathered by a manual annotation, see section 5. In the following sections the three main stages of the algorithm are considered in detail.

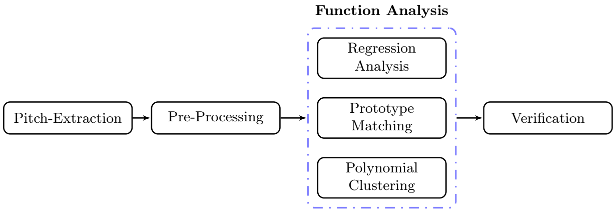


Fig. 1. Structural flowchart of the DP-classification algorithm

3.1 Pitch-Extraction

To extract the pitch-values the phonetic analysis tool “Praat” is used (Boersma [2001]). The pitch estimation is based on auto-correlation. The pitch-value $F0$ for each window is obtained as the fundamental frequency of the speech signal using the autocorrelation function of Praat. To improve the results, the auto-correlation function of the windowed signal $R_{xw}(\nu)$ is divided by the auto-correlation function of the window $R_w(\nu)$, see Eq. 1. This makes the estimation more accurate, noise-resistant and robust than common auto-correlation methods (cf. Boersma [1993]).

$$R_x(\nu) = \frac{R_{xw}(\nu)}{R_w(\nu)} \tag{1}$$

3.2 Pre-Processing

One major problem of a classification using the intonation-curve is the subject of pitch-estimation. As the considered DP “hm” is an unvoiced utterance, it is hard to get a robust pitch estimate. There is no stimulation of the vocal cords at the glottis and thus no “real” pitch is present. This makes it hard to obtain a continuous pitch-contour and can lead to jumps and/or gaps in the contour. To ensure a reliable and robust classification and to reduce the complexity of the function analysis a pre-processing of the original contour is necessary to detect these jumps and gaps. All unsuitable contours are discarded for further investigation on the prototype. To detect these errors of the extraction, the original values are first smoothed by using a one dimensional median filter with a window-size of five samples.

Afterwards, the length of the signal is considered. It is assumed, that at least 50% of the contour needs to be available for the analysis algorithm to reliably assign one of the prototypes (Lotz et al. [2016]). Additionally, for a robust classification the contour should not contain less than ten samples.

Finally, the processing of occurring jumps and gaps is performed. This takes place alternately, beginning with the detection of gaps. A gap is defined as an interruption of the pitch-contour and can be distinguished into two types: A short interruption of the signal for only a few samples and a significant interruption of the signal into two separate sections. The first type can be processed by linear interpolating the missing samples. For the second type the signal is split into two sections, maintaining the longest part for classification. To distinguish between these types of interruptions the ratio between the gap length and the total signal length is calculated. If this ratio exceeds a certain threshold of

$$T_{gap} = 1/3 \tag{2}$$

the interruption is considered as significant. This means, the gap length corresponds to over one third of the signal length. After the processing of gaps, the obtained pitch-contour is now examined on jumps. A jump is defined as a rapid change in the pitch-value, as depicted in Fig. 2. The change can be up to 100Hz but is often split into several consecutive shorter jumps with a smaller difference in the pitch-value. In strongly sloping cases of pitch-contours like DP3 or DP7 (cf. Table 1) these significant changes in pitch-value can also occur. Therefore, the pre-processing algorithm needs to be able to distinguish between these sloping prototypes and jumps. Two methods were developed. It is assumed, that if both methods positively assign a jump to a certain part of the pitch-contour, this assignment is true. For the first method the absolute difference between two consecutive pitch-values $\Delta pitch$ is calculated. If this value exceeds the threshold of

$$\Delta pitch > 10\text{Hz} \tag{3}$$

a jump occurs. The second method evaluates the relative ratio r_{jump} between the absolute jump $\Delta pitch$ and the mean change of pitch in the whole signal. Therefore, the mean rising and falling changes in the pitch need to be distinguished:

$$\overline{\Delta pitch}_{pos} = \frac{1}{P}\sum_{p=1}^{P} \Delta pitch(p); \quad \forall \Delta pitch(i) > 0, \tag{4}$$

$$\overline{\Delta pitch}_{neg} = \frac{1}{N}\sum_{n=1}^{N} \left|\Delta pitch(n)\right|; \quad \forall \Delta pitch(i) < 0. \tag{5}$$

P and N represents the total number of positive and negative $\Delta pitch$ values; p and n are the corresponding sample-indices of i. If there is no differentiation between positive and negative values, by calculating the total mean, these values will cancel each other out. Accordingly, the calculation of r_{jump} is also conducted for both cases. An empirical study has shown that a suitable threshold value to decide if a jump really occurs is

$$r_{jump}(i) = \frac{\left|\Delta pitch(i)\right|}{\overline{\Delta pitch}_{pos/neg}} > 3.5, \tag{6}$$

see Lotz [2014]. If the change in pitch-value exceeds 3.5-times the mean pitch change in positive or negative direction, a jump occurs at the corresponding sample-index i. It is shown that the second method reliably detects jumps in the pitch-contour. For the first method additional assumptions are madeand can be taken from (Lotz [2014]). Furthermore, if a jump is clearly detected, the processing is done similar to the processing of gaps, as described earlier in this section, by splitting the pitch-contour into two sections and retaining the longest part for further processing.

The pre-processing is iteratively repeated until no further gaps or jumps are detected. The remaining pitch-contour is checked again for its signal length and, if applicable,

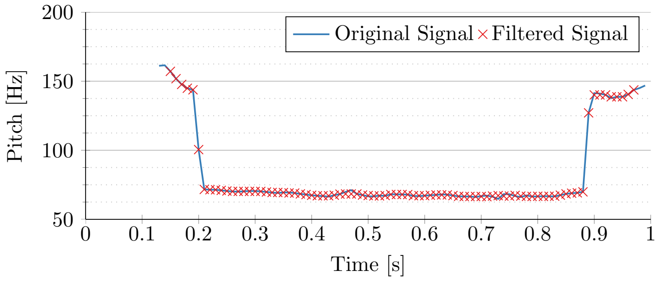


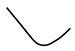
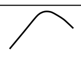
Fig. 2. Original and filtered pitch-contour with two jumps.

forwarded to the function analysis algorithm to assign a prototype.

3.3 Function Analysis

In this stage of the algorithm the pre-processed pitch-contours are assigned to one of the seven form-function-prototypes developed by Schmidt (cf. Table 1). Additional to the original prototypes by Schmidt, two more formtypes DP8 and DP9 (see Table 2) are taken into account. These were frequently noticed in an earlier manual examination of the contours as stated in Lotz [2014]. For these types no information about the functional-meaning is given. This is possible because the classifier only deals with the correct assignment of formtypes and not with their functional-meaning.

Tab. 2. Additional formtypes as stated in Lotz et al. [2015].

Name	Form-Prototype
DP8	
DP9	

Three methods for the assignment of the prototypes are presented in this paper. The original approach is rule-based and uses regression analysis to classify the given pitch-contours. For the second approach mathematical models for each of formtypes were developed and the correlation of the pitch-contours and the modeled formtypes were obtained by calculating their Mean Squared Error (MSE). The third method focused on an unsupervised clustering of pitch-contours into seven classes. This was the only method where no prior knowledge about the formtypes was needed. As a clustering algorithm, simple K-means was applied to the regression coefficients of a third order regression. To enable a reliable clustering a normalization of the regression-curves was done.

4. IN-DEPTH EXPLANATION OF THE UTILIZED FUNCTION ANALYSES

The assignment methods mentioned in the previous section will now be presented in detail.

4.1 Rule-based Regression analysis

The first method for the assignment of the prototypes is based on regression analysis taking advantage of the idealized pitch-contours (cf. Table 1) (Lotz [2014], Lotz et al. [2016]). Only looking at the different course-types they can be distinguished into three types: linear (DP2, DP3 and DP7), quadratic (DP4, DP5, DP8 and DP9) and cubic (DP1, DP6). Depending on the coefficient of determination (R^2) either a first, second or third order regression is performed. If R^2 exceeds the threshold value of

$$R^2 > T_{determination}; \quad T_{determination} = 0.9, \tag{7}$$

it can be assumed that the regression function describes the pitch-contour sufficiently. Furthermore, a tendency of the pitch-contour is determined by considering the slope of the first order regression. It can either be horizontal, rising or falling. This allows the classifier to neglect some prototypes for further assignment. For example, if a pitch-contour has a falling tendency, all prototypes with a clear horizontal (DP2) or rising tendency (DP4, DP7 or DP9) can be neglected. Only prototypes with a rising slope tendency are considered as possible assignment.

Linear Regression If a pitch-contour is sufficiently described by a linear regression it can be unambiguously assigned to a linear course-type by looking at its slope tendency. If the absolute slope exceeds a threshold of

$$T_{slope} = 40\text{Hz/s}, \tag{8}$$

the course is assumed to be either rising or falling (depending on the sign). Otherwise, it is assumed to be horizontal. Additionally, for horizontal regression lines, disregarding threshold (7), the standard deviation of the signal is taken into account. If the standard deviation of the original course is less than 7Hz, type DP2 is assigned. Variations of the speech signal of this small size can be neglected, as the human ear is not able to perceive them (Wendemuth [2004]).

Higher Order Regression In all cases in which the pitch-contour cannot be described by a linear regression function, the regression order is increased. The highest order of regression is a third order fitting. Previous research has shown, that only for a very small amount of the investigated data it was necessary to perform a higher order regression to satisfy condition (7) (cf. Lotz [2014]). The general approach for higher order regression functions is stated as followed: First, the pitch-contour is divided into segments limited by their turning points. Then the slope of the segments is determined. As stated in the section about linear regression, the segments are assumed to be sloping when exceeding threshold T_{slope}. This makes it possible to already get a tendency of the contour. Depending on this tendency either the slope-ratio or time-ratio of two consecutive segments is calculated and used to determine the prototype. The thresholds for these parameters are:

$$T_{sloperatio} = T_{timeratio} = 1/3. \tag{9}$$

If the segment exceeds this value, the respectively "shorter" segment is left unconsidered, only regarding the dominant part for the prototype assignment. Otherwise, both segments will be taken into account (cf. Lotz et al. [2016]).

4.2 Prototype Matching

For the second approach a mathematical model for each considered form-function prototype by Schmidt was developed. Simple linear, quadratic and sine functions were used. For the DP1 no model was implemented as in the manual labeling process (see section 6.1) less than 1% of the pitch-contours were assigned to this prototype. For DP2, DP3 and DP7 simple linear functions were implemented:

$$y_{DP2}(x) = \overline{pitch}, \tag{10}$$

$$y_{DP3}(x) = -n \cdot x + n, \tag{11}$$

$$y_{DP7}(x) = n \cdot x. \tag{12}$$

For all mentioned models x runs from $-\pi$ to $+\pi$; y_i represent the values of the modeled pitch-contours, were i

denotes the modeled prototype (DP2, DP3, DP4, DP5, DP6, DP7). To make the classifier more accurate, not only the idealized contours were implemented but also intermediate models with $n = [60, 80, 100, 200, 300, 400]$. To allow small rising and falling slopes for DP2, intermediate models of DP2 are generated from DP3 and DP7 but having a smaller slope of only $n = [10, 20, 30, 40]$. The mathematical models for the remaining prototypes are calculated as followed:

$$y_{DP4}(x) = x^2, \tag{13}$$

$$y_{DP5}(x) = -x^2, \tag{14}$$

$$y_{DP6}(x) = \sin(x). \tag{15}$$

For DP4 and DP5 certain sub-parts of the resulting functions constituting the desired prototype are used.

To be able to calculate the MSE the constructed contour-models need to correlate with the extracted pitch-contour. This is done by interpolating the models to the time-values of the pitch-contours. Then the model is moved towards the contour by subtracting the maximum value of the interpolated signal and adding the maximum pitch value to the model. This will lead to concurring starting/ending points of the prototype model and the pitch-contour.

An example for a linear prototype matching is depicted in Fig. 3. In the upper subplot the red line represents the original non-processed pitch-contour. After pre-processing only the samples depicted as blue crosses, remained for the prototype assignment. The lower subplot pictures the prototype match. By calculating the MSE, DP2 was chosen to be the desired prototype.

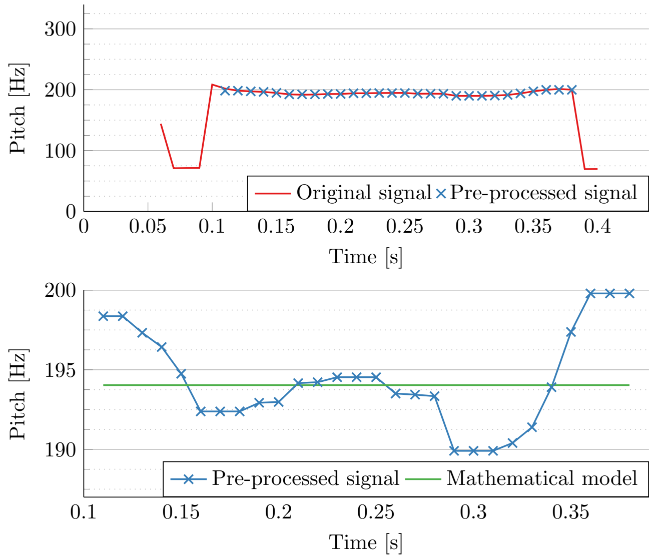


Fig. 3. Upper subplot: original and pre-processed pitch-contour of type DP2; Lower subplot: pre-processed pitch contour and claimed prototype-model also resulting in a match in type DP2

To calculate the MSE, the models are interpolated and both (model and contour) are normalized to the values $[0, 1]$ in x/time and y/pitch direction, then y and x of the model are rescaled to the pitch/time-range of the pre-processed contour. The MSE is calculated for every prototype and the model with the lowest MSE-value is chosen to be the claimed prototype. Fig. 4 pictures a prototype matching for a pitch-contour of type DP4. As in Fig. 3 the red line in the upper subplot represents the original non-processed pitch-contour, the blue crosses represent the remaining samples after the pre-processing. The lower subplot depicts the matched prototype, DP4.

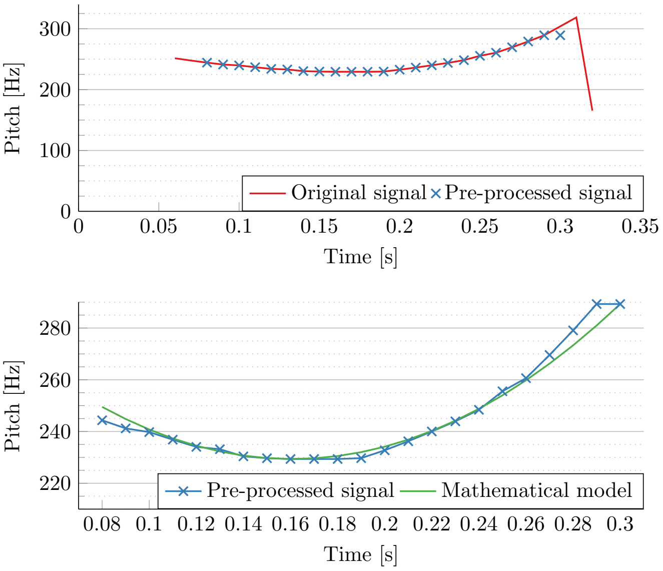


Fig. 4. Upper subplot: Original and pre-processed pitch-contour of type DP4; Lower subplot: Pre-processed pitch-contour and claimed prototype-model also resulting in a match in type DP4

4.3 K-means regression parameter clustering

This approach focuses on an unsupervised clustering of pitch-contours into several classes, according to the occurring form-function prototypes by Schmidt. A similar method to cluster foot-based pitch contours is described in Klabbers and van Santen [2004]. The authors assume that pitch curves are tied to the foot structure of a sentence, whereas a foot is defined as consisting of an accented syllable followed by all unaccented syllables that precede the next accented syllable or a phrase boundary (cf. Thorsen [1980]). As the authors only investigate contours of falling or rising pitch tendencies or contours containing peaks, they directly work on the normalized pitch-contours. Considering the investigation presented in this section this is not possible, as for the investigation presented in this paper also alternating contours need to be considered. This is done by using the coefficients of a polynomial regression for the following clustering. First, the data samples are normalized. All contours are centered on the time axis (X_{cen}) and the pitch values are mean-adjusted (Y_{adj}):

$$X_{cen} = (X - \mu)/\sigma, \tag{16}$$

$$Y_{adj} = Y - \mu. \tag{17}$$

These centering and scaling transformations improve the numerical properties of the calculation of the regression coefficients, see Klabbers and van Santen [2004]. Afterwards, a third order regression function is calculated:

$$y = ax^3 + bx^2 + cx + d. \tag{18}$$

From an inspection of the used datasets, and due to the knowledge that according to Schmidt a third order polynomial (DP6, see Table 1) needs to be present, polynomials of cubic order were used. Polynomials of higher order were not used, to avoid over-fitting.

The four obtained coefficients (a, b, c, d) were used to cluster the pitch-contours into the different classes, assuming that for all prototypes considered by Schmidt samples are present in the train-set. For the clustering a k-means clustering was utilized. This is an iterative clustering method for finding membership of observations, where the model depends on unobserved latent variables (cf. Klabbers and van Santen [2004]). A requirement for this algorithm is the pre-defined number of clusters k. This number is used to initialize k clusters and iteratively find their cluster-centers. k is chosen as the number of formtypes identified during the manual labeling.

5. MANUAL LABELING FOR VERIFICATION

To ensure the reliability of the classification-algorithm and verify the results a statement about how well the obtained prototypes match the original signals is needed. This statement is gathered by comparing the output of the classification-algorithm with a ground truth on the course of the DP-samples. Therefore, a manual labeling of the pitch-contour is performed (cf. Lotz et al. [2015], Lotz et al. [2016]).

For the manual labeling a visual assignment of the pitch-contours to the idealized formtypes developed by Schmidt (cf. Table 1) is performed. No information about the surrounding content or the speech material itself is given to the labelers. This permits an independent objective examination of the pitch-contours. Two rounds of independent labeling were performed, one for the development dataset (LMC) based on HCI and one for the evaluation dataset (ALICO) based on HHI.

For LMC, in total 8 labelers were asked to assign the depicted 259 pitch-contours to either one of the seven formtypes by Schmidt, one of the two additional formtypes (DP8, DP9), or the option “no specification possible” (NSP) to the depicted pitch-contour. Thereby, the last option should only be considered if clearly none of the nine formtypes match.

For ALICO, in total 10 labelers, that took not part in the labeling of the LMC samples, were asked to assign the depicted 537 pitch-contours to either one of the seven form-function-prototypes by Schmidt or the option “no specification possible” (NSP). For this labeling, the additional formtypes DP8 and DP9 were left unconsidered on purpose, as an earlier investigation of the functional-meaning of these formtypes has shown, that they all state similar descriptions as DP4 and DP6 (confirmation and positive assessment → positive feedback). Therefore, these formtypes were generalized into DP6, as their contours are also just subsets of the contour of DP6 (cf. Lotz et al. [2016]).

6. RESULTS

In this section the results of the manual labeling and the classification results of the three described methods are presented.

6.1 Manual Labeling

For both dataset LMC and ALICO a majority voting is conducted. To secure the reliability of the labeling Krippendorf’s α is calculated. In case of the LMC dataset the results of all labelers were taken into account for the majority voting. If more than five labelers agreed on one formtype, this formtype was assigned to the pitch-contour. The inter-rater reliability was calculated to a value of $\alpha_{LMC} = 0.53$. For the ALICO dataset the five labelers with the highest inter-rater reliability $\alpha_{ALICO} = 0.64$ were chosen for the majority voting. On LMC a reduction of labelers does not increase Krippendorf’s α.

The results of the majority voting for the considered data-samples (received from the pre-processing, see section 3.2) are stated in Fig. 5. Only data-samples rated as suitable by the pre-processing step ($N_{LMC} = 128$, $N_{ALICO} = 246$) are used for the manual labeling and thus depicted, as only these samples are later considered to calculate the performance of the three different classification methods. As stated in the introduction (cf. section 1), an increase of partner-oriented feedback signals in case of HHI compared to HCI can be assumed. This is confirmed by the results of the manual labeling, with an increase of formtype DP4 (confirmation) and a decrease of the speech-organizing formtype DP2 (thinking) for the ALICO corpus (HHI). Furthermore, for the ALICO data-samples no pitch-contours are assigned to the formtypes DP8 and DP9, as these formtypes were not available to the labelers. The pitch-contours for which no majority of the labeling was achieved are assigned to the class “NM”. These samples were not taken into account to calculate the performance of the classifier, as no statement about the correctness of the assignment is possible.

In total, for the further investigation, 113 data-samples of LMC and 218 data-samples of the ALICO corpus are available for the classifier comparison.

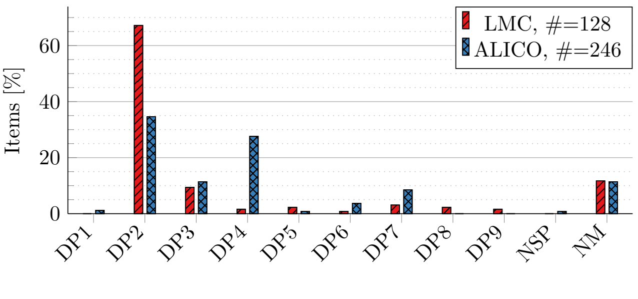


Fig. 5. Frequency distribution of the majority voting on both corpora, LMC and ALICO. Additionally used labels: NSP “no specification possible.” and NM “no majority”

6.2 Rule-based Regression analysis

In case of the rule-based approach, the seven form-function prototypes by Schmidt as well as the additional formtypes

DP8 and DP9 were used to be assigned by the classifier, as for the manual labeling of the LMC corpus also all nine formtypes were available and assigned during the labeling. Table 3 shows the confusion matrix of the rule-base regression analysis algorithm compared to the manual labeling of the LMC corpus. The main diagonal depicts the number of pitch-contours correctly assigned by the classifier. All other occupied entries state mismatches in the assigned formtypes. For the calculation of the performance of this classifier all 113 pitch-contours were taken into account, resulting in a correspondence of 89.4% of the classifier and the labeling. Concerning the total number of contours no significant number of mismatches in specific formtypes is recognized.

Tab. 3. Confusion matrix of the classifier based on regression analysis using the manual labeling (LMC) as ground truth.

		Manual Labeling									
		DP1	DP2	DP3	DP4	DP5	DP6	DP7	DP8	DP9	NSP
Classifier	DP1	-	-	-	-	-	-	-	-	-	-
	DP2	-	78	-	-	-	1	-	-	-	-
	DP3	-	1	11	-	-	-	-	-	-	-
	DP4	-	-	-	2	-	-	-	-	1	-
	DP5	-	1	-	-	3	-	-	-	-	-
	DP6	-	1	-	-	-	-	-	-	-	-
	DP7	-	1	-	-	-	-	4	-	1	-
	DP8	-	2	1	-	-	-	-	3	-	-
	DP9	-	2	-	-	-	-	-	-	-	-

For the ALICO corpus only the seven form-function prototypes by Schmidt were available after the labeling process and utilized to test the developed classifier. As the labelers had no option to assign prototypes of DP8 or DP9, they were not taken into account, leaving 194 data-samples to calculate the performance of the classifier. The performance of the classifier is as well calculated only on these seven formtypes. Table 4 shows the confusion matrix of the rule-based classification algorithm compared to the results of the labeling. A consistency of the results was shown in 81.4% of the considered data-samples.

The highest mismatch can be seen in the assignment of formtype DP3 by the classifier. Considering the results of earlier investigations, it is typical that labelers more likely assigned formtypes with a horizontal tendency than sloping formtypes (cf. Lotz [2014]). Also the confusion of the formtypes DP4 and DP7 is expectable, as the examination of the functional-meaning of these formtypes has shown a high mismatch (cf. Lotz et al. [2016]). Instead of a clear linear course a slightly convex contour is perceived by the human labelers.

Tab. 4. Confusion matrix of the classifier based on regression analysis using the manual labeling (ALICO) as ground truth.

		Manual Labeling							
		DP1	DP2	DP3	DP4	DP5	DP6	DP7	NSP
Classifier	DP1	-	-	-	-	-	-	-	-
	DP2	1	65	-	6	-	1	-	-
	DP3	2	8	16	-	-	1	-	-
	DP4	-	4	-	56	-	1	5	-
	DP5	-	1	-	-	1	-	-	-
	DP6	-	-	-	4	-	6	-	-
	DP7	-	2	-	-	-	-	14	-

6.3 Prototype Matching

For the prototype matching the formtypes DP8 and DP9 were left unconsidered by the classifier. As the labeling of the LMC data-samples was conducted in an earlier investigation, the labelers had the option to assign these two additional formtypes. Accordingly, there will never be a match of DP8 or DP9 between the manually annotated labels and the results of the classifier. They will therefore not be taken into account to verify the performance of this classification method. Disregarding the assignment of formtypes DP8 and DP9 by the labelers this leads to 108 data-samples for LMC. The classification achieves a consistency of 83.3%. Table 5 shows the confusion matrix of the prototype matching and the results of the manual labeling.

Tab. 5. Confusion matrix of the classifier based on mathematical models of the prototypes using the manual labeling (LMC) as ground truth.

		Manual Labeling							
		DP1	DP2	DP3	DP4	DP5	DP6	DP7	NSP
Classifier	DP2	-	72	-	-	1	1	-	-
	DP3	-	4	12	-	1	-	-	-
	DP4	-	4	-	2	-	-	-	-
	DP5	-	3	-	-	1	-	-	-
	DP6	-	2	-	-	-	-	1	-
	DP7	-	1	-	-	-	-	3	-

The highest mismatch is recognized for formtype DP2. This is explainable by the high sensitivity of the classifier towards sloping pitch-contours as depicted in Fig. 6. Because of the normalization and rescaling of the mathematical model, small changes in the value of the pitch-contour are identified as significant changes and are therefore falsely assigned to a quadratic formtype.

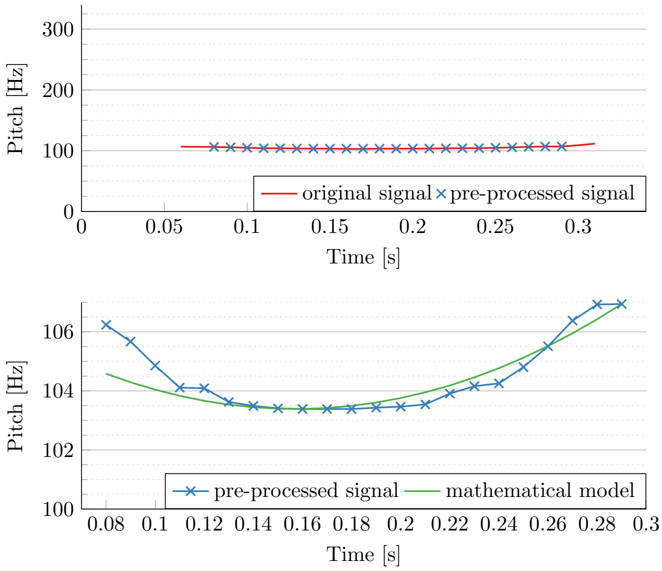


Fig. 6. Upper subplot: Original and pre-processed pitch-contour of type DP2; Lower subplot: Pre-processed pitch-contour and claimed prototype-model resulting in a mismatch of DP4

In the manual labeling of the ALICO data-samples the formtypes DP8 and DP9 were not available to the labelers.

As the classifier also discards these formtypes, all 218 data-samples are used for further investigation. The considered samples led to a consistency of the results in 56.9% of all cases, excluding the formtype DP1 and all cases where no specification was possible by the labelers (denoted as NSP). As before, Table 6 shows the confusion matrix of the prototype matching compared to the results of the manual labeling. As mentioned in the results of the rule-based classifier a high mismatch is recognized in case of the assignment of the classifier to the formtype DP2. This is on the one hand explicable by the results mentioned in Lotz [2014]: the labelers are more likely assigning horizontal prototypes than sloping ones. On the other hand this is expectable by the high sensitivity of the present approach towards polynomial formtypes (cf. Fig. 6). This issue was already discussed for the mismatch of samples in the LMC data.

Tab. 6. Confusion matrix of the classifier based on mathematical models of the prototypes using the manual labeling (ALICO) as ground truth.

		Manual Labeling							
		DP1	DP2	DP3	DP4	DP5	DP6	DP7	NSP
Classifier	DP2	-	36	4	8	-	4	-	-
	DP3	2	14	23	1	-	1	-	1
	DP4	-	24	1	47	-	1	6	-
	DP5	1	6	-	-	1	1	-	-
	DP6	-	3	-	8	1	2	-	1
	DP7	-	2	-	4	-	-	15	-

6.4 Polynomial Fitting and k-means clustering

In this section the results of the k-means clustering of the normalized third order regression parameters are presented. As stated in section 4.3, for this approach the number of desired clusters k needs to be defined beforehand. Regarding the seven form-function relations by Schmidt and the results of the manual labeling, the number of clusters, for the LMC, was set to six. For DP1 no cluster was generated, as it was never assigned by the labelers. After choosing the number of clusters, a manually assignment which cluster represents a certain formtype was carried out, using the results of the labeling of the LMC dataset (multiple selections possible). The resulting cluster assignments were then applied to the ALICO data-samples. As for the "prototype matching"-classifier the formtypes DP8 and DP9 were left unconsidered in both datasets, leaving 108 samples of the LMC and 218 samples of the ALICO corpus to calculate the performance of this classification method.

As the presented method is an unsupervised k-means clustering, the outcome of the clustering contains no direct assignment of the clusters to the formtypes. Therefore, the outcome of the classifier needs to be predicted by assigning the resulting clusters, denoted as Cn with $n = 1, ..., 6$, to the formtypes by Schmidt. This was done by inspecting the assignment matrix pictured in Table 7. It shows the agreement between the results of the k-means clustering and the results of the manual labeling of the LMC data-samples. The clusters with the highest agreement in one formtype were assigned to this formtype. The resulting cluster-formtype assignments are tagged in different colors. These clusters were later used to assess the samples of the ALICO data as well.

Tab. 7. Assignment matrix of the k-means algorithm using the manual labeling (LMC) as ground truth.

		Manual Labeling							
		DP1	DP2	DP3	DP4	DP5	DP6	DP7	NSP
k-means	C1	-	33	3	-	-	1	3	-
	C2	-	-	-	-	-	-	1	-
	C3	-	1	1	2	-	-	-	-
	C4	-	23	3	-	1	-	-	-
	C5	-	29	5	-	-	-	-	-
	C6	-	-	-	-	2	-	-	-

Considering Table 7 this leads to the following assignments: DP2 is clearly assembled from C1, C4 and C5 (orange), as the agreement between these clusters and DP2 is dominating all other agreements of the possible assignments. DP4, DP5 and DP7 can each be unambiguously assigned to one cluster: DP4 is represented by C3 (blue), DP5 by C6 (purple) and DP7 by C1 (green). For DP3 and DP6 no distinct cluster assignment is possible. As DP1 was not represented in the results of the manual labeling, no cluster is assigned to this formtype. Following this cluster-formtype assignment, the classifier achieved a consistency of 83.3% between the results of the manual labeling and the clustering on the LMC data-samples.

The resulting cluster assignments on the LMC data-samples are now applied to the data-samples of the ALICO corpus (cf. Table 8). As mentioned earlier in this section only the valid pre-processed samples of the ALICO samples are clustered ($N_{ALICO} = 218$).

Tab. 8. Assignment matrix of the k-means algorithm using the manual labeling (ALICO) as ground truth.

		Manual Labeling							
		DP1	DP2	DP3	DP4	DP5	DP6	DP7	NSP
k-means	C1	3	19	2	1	1	2	9	-
	C2	-	-	-	1	-	3	-	1
	C3	-	1	2	47	-	-	-	-
	C4	-	8	7	1	1	2	2	-
	C5	-	57	17	18	-	2	10	-
	C6	-	-	-	-	-	-	-	1

For the ALICO data (cf. Table 8), the same cluster assignments as for the LMC dataset were applied: DP2 is assembled from C1, C4 and C5 (orange), DP4 is represented by C3 (blue), DP5 by C6 (purple) and DP7 by C1 (green). For DP1, DP3 and DP6 no cluster assignment is possible, as no cluster assignment of the LMC dataset exists for these formtypes. Following this cluster-formtype assignment, the classifier achieved a consistency of 60.1% on the ALICO data. This is a remarkably lower consistency than for the LMC data. But it has to be mentioned that the formtypes DP1 and DP6 are not or just rarely represented in the LMC training samples. As only four out of seven prototypes could unambiguously be assigned to the resulting clusters of the LMC dataset, all other prototypes will not be correctly assigned to the data-samples of the ALICO corpus. This means 17.9% out of the 218 considered data-samples will automatically mismatch.

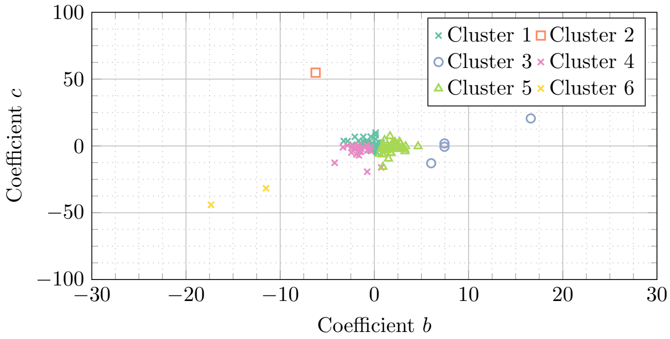


Fig. 7. Distribution of resulting clusters on LMC, using k-means, plotted on the b-c-plane.

To explain why the clustering on the LMC data achieved a higher performance than the clustering on the ALICO data, the resulting clusters are represented in two dimensions (regression coefficient c against coefficient b) for both datasets, see Fig. 7 (LMC - HCI) and Fig. 8 (ALICO - HHI). It can be seen that for the LMC data, the coefficients of the polynomial fitting are more dense than for the ALICO corpus. Additionally, cluster 2 and cluster 6 are clearly separated from all other clusters but also only contain a few data-samples. It can be assumed that these data-samples are just outliers, not typically used in HCI. This assumption is also confirmed by the fact that the corresponding clusters 2 and 6 are not represented as DP7 and DP5 in the ALICO corpus. Not considering these outliers for the LMC, the values in Fig. 7 range from -4.2 to 16.6 (standard deviation of 2.6) for the coefficient b and -19.3 to 20.6 (standard deviation of 5.1) for the coefficient c. For the ALICO corpus (cf. Fig. 8), the values for b range from -32.9 to 17.0 (standard deviation of 4.5) and for c from -18.1 to 34.3 (standard deviation of 9.4), respectively.

It can be assumed that the more scattered coefficients of the ALICO samples are due to the fact, that the way DPs are uttered is influenced by the conversational partner. Especially in HCI the human partners tend to articulate differently than in human communications (cf. McKeown et al. [2012], Yu et al. [2004]). Thus the amplitudes on the contours are different as well, due to the naturalness of the data and the presence of DP4, which is nearly not present in the HCI data.

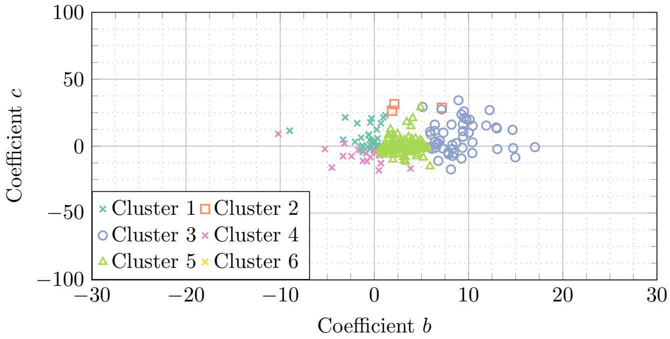


Fig. 8. Distribution of resulting clusters on ALICO, using the same k-means as for LMC, plotted on the b-c-plane.

7. CONCLUSION

Comparing the results of the different classification algorithms, the best results for both datasets (HCI LMC = 89.4% and HHI ALICO = 81.4%) could be obtained using the rule-based approach (cf. Table 9). The prototype matching and k-means regression parameter clustering achieved the same results for the LMC dataset. For the ALICO corpus the results of the prototype matching were slightly better than the results of the k-means clustering. This is expectable as the ALICO data contained formtypes not considered in the LMC corpus. This also supports the assumption that HCI and HHI datasets have a different distribution of partner oriented feedback signals (cf. Fischer et al. [1996]).

Tab. 9. Comparison of the achieved performances of all developed classification algorithms for LMC and ALICO.

	Performance [%]	
Method	LMC	ALICO
Regression Analysis	89.4	81.4
Prototype Matching	83.3	56.9
Polynomial Clustering	83.3	60.1

To further improve the performance of all classifiers a balanced distribution of all form-function-prototypes should be ensured. As the investigation so far only deals with the classification of “isolated” DPs, in terms of no surrounding content of the considered DPs, the desired dataset can be merged from different corpora of the same conversational style (HCI/HHI) and level of naturalness (naturalistic/acted). Having these more balanced data, sophisticated classification methods, such as decision-tree based algorithms or random forests, can be trained.

The consideration of the form-function-relation of feedback signals enables technical systems to examine longer-lasting natural interactions and dialogues, and to evaluate the user’s feedback. This leads to a more natural human-computer interaction. Technical systems that use these extended skills behave more natural and thus become the user’s attendant and ultimately his companion (cf. Biundo and Wendemuth [2015]).

ACKNOWLEDGEMENTS

The work presented was done within the Transregional Collaborative Research Centre SFB/TRR 62 “Companion-Technology for Cognitive Technical Systems” (www.sfb-trr-62.de) funded by the German Research Foundation (DFG).

REFERENCES

J. Allwood, J. Nivre, and E. Ahlsén. On the semantics and pragmatics of linguistic feedback. *Journal of Semantics*, 9:1–26, 1992.

P. Baranyi, A. Csapo, and G. Sallai. *Cognitive Infocommunications (CogInfoCom)*. Springer, Berlin, Germany, 2015.

L. Bächler, A. Bächler, M. Kölz, T. Hörz, and T. Heidenreich. Über die Entwicklung eines prozedural-interaktiven

Assistenzsystems für leistungsgeminderte und gewandelte Mitarbeiter in der manuellen Montage. *Kognitive Systeme*, 1:s.p., 2015.

B. Benninghoff, P. Kulms, L. Hoffmann, and N. C. Krämer. Theory of mind in human-robot-communication: Appreciated or not? *Kognitive Systeme*, 1:s.p., 2013.

S. Benus, A. Gravana, and J. Hirschberg. The Prosody of Backchannels in American Englisch. In *Proc. of the 16th ICPhS*, pages 1065–1068, Saarbrücken, Germany, 2007.

S. Biundo and A. Wendemuth. Companion-technology for cognitive technical systems. *KI - Künstliche Intelligenz*, 30(1):71–75, 2015.

P. Boersma. Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound. *Proc. of the Institute of Phonetic Sciences*, 17:97–110, 1993.

P. Boersma. Praat, a system for doing phonetics by computer. *Glot International*, 5(9-10):341–345, 2001.

H. Buschmeier, Z. Malisz, J. Skubisz, M. Wlodarczak, I. Wachsmuth, S. Kopp, and P. Wagner. ALICO: a Multimodal Corpus for the Study of Active Listening. In *Proc. of the 9th LREC*, Reykjavik, Iceland, May 2014.

M. Corley and O. W. Stewart. Hesitation Disfluencies in Spontaneous Speech: The Meaning of *um*. *Language and Linguistics Compass*, 2:589–602, 2008.

K. Fischer. *From Cognitive Semantics to Lexical Pragmatics.* Mouton & Co, Berlin, New York, 2000.

K. Fischer, B. Wrede, C. Brindöpke, and M. Johanntokrax. Quantitative und funktionale Analysen von Diskurspartikeln im Computer Talk (Quantitative and functional analyzes of discourse particles in Computer Talk). *Int. Journal for Language Data Processing*, 20:85–100, 1996.

J. Frommer, B. Michaelis, D. Rösner, A. Wendemuth, R. Friesen, M. Haase, M. Kunze, R. Andrich, J. Lange, A. Panning, and I. Siegert. Towards emotion and affect detection in the multimodal last minute corpus. In *Proc. of the 8th LREC*, pages 3064–3069, Istanbul, Turkey, 2012.

R. Kehrein and S. Rabanus. Ein Modell zur funktionalen Beschreibung von Diskurspartikeln (A Model for the functional description of discourse particles). In *Neue Wege der Intonationsforschung*, volume 157-158 of *Germanistische Linguistik*, pages 33–50. Georg Olms Verlag, Hildesheim, Germany, 2001.

E. Klabbers and J. P. H. van Santen. Clustering of foot-based pitch contours in expressive speech. In *Proc. of the 5th ISCA Workshop on Speech Synthesis*, pages 73–78, Pittsburgh, PA, USA, 2004.

R. D. Ladd. Intonational Phonology. In *Studies in Linguistics*, volume 79. Cambridge University Press, Cambridge, UK, 1996.

A. F. Lotz. Differentiation von Form-Funktions-Verläufen des Diskurs Partikels "hm" über unterschiedliche mathemathische Herangehensweisen (Differentiation of form-function-relations of the discourse particle "hm" using different mathematical approaches). Master's thesis, Otto–von–Guericke University Magdeburg, 2014.

A. F. Lotz, I. Siegert, and A. Wendemuth. Automatic differentiation of form–function–relations of the discourse particle “hm” in a naturalistic human-computer interaction. In Wirsching G., editor, *Elektronische Sprachsignalverarbeitung 2015*, volume 78 of *Studientexte zur Sprachkommunikation*, pages 172–179. TUDpress, 2015.

A. F. Lotz, I. Siegert, and A. Wendemuth. Classification of functional-meanings of non-isolated discourse particles in human-human-interaction. In *Human-Computer Interaction. Theory, Design, Development and Practice*, volume 9731 of *LNCS*, pages 53–64. Springer, 2016.

G. McKeown, M. Valstar, R. Cowie, M. Pantic, and M. Schroder. The SEMAINE Database: Annotated Multimodal Records of Emotionally Colored Conversations between a Person and a Limited Agent. *IEEE Trans. Affect. Comput.*, 3:5–17, 2012.

H. Paschen. Die Funktion der Diskurspartikel HM (The function of discourse particles HM). Master's thesis, University Mainz, 1995.

D. Rösner, M. Kunze, M. Otto, and J. Frommer. Linguistic analyses of the LAST MINUTE corpus. In Jeremy Jancsary, editor, *Proc. of KONVENS 2012*, pages 145–154. ÖGAI, September 2012.

D. Rösner, M. Haase, T. Bauer, S. Günther, J. Krüger, and J. Frommer. Desiderata for the Design of Companion Systems – Insights from a Large Scale Wizard of Oz Experiment. *Künstliche Intelligenz*, 30(1):53–61, 2015.

J. E. Schmidt. Bausteine der Intonation (Components of intonation). In *Neue Wege der Intonationsforschung*, volume 157-158 of *Germanistische Linguistik*, pages 9–32. Georg Olms Verlag, Hildesheim, Germany, 2001.

I. Siegert, R. Böck, and A. Wendemuth. The influence of context knowledge for multimodal annotation on natural material. In *Joint Proceedings of the IVA 2012 Workshops*, pages 25–32, Santa Cruz, USA, 2012.

I. Siegert, K. Hartmann, D. Philippou-Hübner, and A. Wendemuth. Human Behaviour in HCI: Complex Emotion Detection through Sparse Speech Features. In A. Salah, H. Hung, O. Aran, and H. Gunes, editors, *Human Behavior Understanding*, volume 8212 of *LNCS*, pages 246–257. Springer, Berlin, Germany, 2013.

I. Siegert, D. Philippou-Hübner, K. Hartmann, R. Böck, and A. Wendemuth. Investigation of speaker group-dependent modelling for recognition of affective states from speech. *Cognitive Computation*, 6(4):892–913, 2014a.

I. Siegert, D. Prylipko, K. Hartmann, R. Böck, and A. Wendemuth. Investigating the form-function-relation of the discourse particle “hm” in a naturalistic human-computer interaction. In S. Bassis, A. Esposito, and F. Morabito, editors, *Recent Advances of Neural Network Models and Applications*, volume 26 of *Smart Innovation, Systems and Technologies*, pages 387–394. Springer, Berlin, Germany, 2014b.

G. Skantze, M. Johansson, and J. Beskow. Exploring turn-taking cues in multi-party human-robot discussions about objects. In *Proc of the 2015 International Conference on Multimodal Interaction*, pages 67–74, 2015.

N. Thorsen. Intonation contours and stress group patterns in declarative sentences of varying length in asc danish. *Annual Report*, 14:13–47, 1980.

A. Wendemuth. *Grundlagen der stochastischen Sprachverarbeitung.* Oldenbourg, Munich, Germany, 2004.

C. Yu, P. M. Aoki, and A. Woodruff. Detecting user engagement in everyday conversations. In *Proc. of the INTERSPEECH-2004*, pages 1329–1332, Jeju, Korea, 2004.