Interspeech 2025
17-21 August 2025, Rotterdam, The Netherlands

Continuous prediction of backchannel timing for human-robot interaction

Michael Paierl, Martin Hagmüller, Barbara Schuppler

Signal Processing and Speech Communication Laboratory, Graz University of Technology, Austria
{paierl, hagmueller, b.schuppler}@tugraz.at

Abstract

This paper explores the continuous prediction of backchannel timing in conversational speech, with the aim to make turn-taking in human-robot interaction more natural. To assure real-time prediction, we present regression-based models based exclusively on acoustic features that can be extracted continuously from the user's speech. Comparing different machine learning models, we found lightGBM models to perform best with respect to accuracy (mean absolute error: approx. 130 ms) and efficiency, while meeting the real-time requirement. Our analysis of feature importances revealed that speaking duration, intensity and fundamental frequency are among the most important predictors of backchannel timing, when extracted in the window from 275-875 ms before a backchannel in the interlocutor's turn. Given the strong predictive performance of our models, this work provides a foundation for implementing more natural and responsive conversational agents.
Index Terms: backchannels, human-robot interaction, turn-taking, prosodic features

1. Introduction

The integration of social robots into our daily lives has advanced rapidly, with robots increasingly being deployed in domains such as education [1] and healthcare [2] to customer service [3]. A key aspect of natural conversation is backchanneling, a short listener response like "mhm", "yeah" or head nods, which play a crucial role in signaling understanding, engagement, and maintaining conversational flow [4]. Despite progress in conversational AI, the ability to produce backchannels in a timely and contextually appropriate manner remains an open challenge. In this paper, we focus on both the timing and the conversational context of backchanneling for human-robot interaction (HRI). For this purpose, we investigated a real-time approach for predicting the timing of backchannels while using manually created turn-taking annotations to control for different communicative functions.

Backchanneling is not only of interest to researchers in HRI, but also to speech scientists aiming at increasing our understanding of speech planning and processing in talk in interaction. As grounding mechanisms, backchannels help ensure mutual understanding in a conversation [5]. Boegels et al. [6] found that cues of an upcoming backchannel often include a falling or flat intonation, pauses, and a reduction in articulation rate (final lengthening) in the interlocutor's speech. Most existing studies focus solely on acoustic patterns or listener behavior without considering the communicative, turn-taking function of the speech produced by the interlocutor that precedes the backchannel. Unlike studies that rely on orthographic annotations such as "mhm", which may not always indicate whether that token actually functions as a backchannel, this paper uses turn-taking annotations [7] that were manually created following a conversation analysis (CA) approach [8]. This enables more precise analysis of backchannel–turn-taking interaction.

Incorporating backchannels into HRI poses unique challenges. Social robots, like Furhat [9], aim to simulate human-like conversational behaviors using multimodal cues, including expressive faces and speech [10, 11, 12]. However, a core difficulty lies in timing backchannels accurately in real time. This involves two distinct tasks: (1) determining whether a backchannel should occur at all and (2) predicting the precise timing of its occurrence. While early rule-based models approached backchannel prediction with handcrafted heuristics [13], advances in deep learning have enabled data-driven approaches. Recent work has explored Transformer-based architectures [14, 15], with the Voice Activity Projection (VAP) model emerging as a promising tool for turn-taking prediction [16]. Given the close relationship between turn-taking dynamics and backchannel behavior, the VAP model has been fine-tuned for continuous real-time backchannel timing prediction and backchannel type prediction in Japanese [17].

Despite these advancements, backchannel prediction in HRI requires a careful balance. A simple approach, such as inserting a backchannel after every pause, results in excessive feedback, leading to unnatural interactions [18]. Predicting whether a backchannel is necessary is inherently complex, as backchannels are optional and not obligatory. Their frequency varies depending on the conversational strategies of the speaker and listener. Therefore, this study focuses solely on the continuous prediction of backchannel timing, assuming the necessity of a backchannel is already determined. This task requires analyzing preceding speech cues in real time while minimizing latency. While Transformer-based models like VAP offer powerful sequence-learning capabilities, their black-box nature obscures the influence of individual features (e.g., intensity). This hinders interpretability, a key requirement for controllable HRI.

To address these challenges, this paper explores real-time backchannel timing prediction through machine learning while maintaining interpretability of feature impact[1]. Unlike many existing approaches, this study emphasizes efficiency and practicality for real-world applications. By analyzing prosodic features such as speaking duration, intensity, and F0, alongside turn-taking functions, we investigate the feasibility of real-time prediction with a focus on balancing model performance and interpretability. These findings contribute to the broader goal of developing conversational agents that can engage in more natural, fluid, and human-like dialogue, paving the way for further advancements in HRI technology.

[1]This work builds upon the master thesis of the first author [19].

10.21437/Interspeech.2025-770

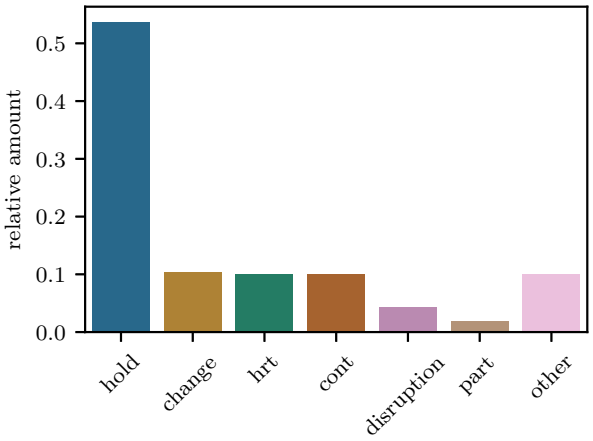

Figure 1: *Relative distribution of the six most common turn-taking functions of the PHRTs.*

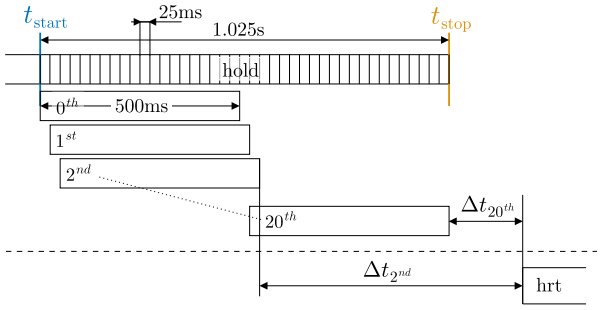

Figure 2: *Illustration of the windowing process of the PHRT tokens. The windows have size $N = 500$ ms with a hop size $R = 25$ ms. The targets (i.e., PHRT HRT time gap) can be seen as $\Delta t_{i^{th}}$.*

2. Materials

2.1. GRASS corpus

In this paper, we use the conversational speech component of the GRASS corpus [20, 21], which consists of 19 dyadic one-hour long spontaneous conversations between close friends or family members in Austrian German. The dialogues were recorded in a studio without the presence of an experimenter and were unscripted and topic free, ensuring natural interactions with a rich variety of backchanneling behavior. The corpus comes with manually created orthographic transcriptions and turn-taking annotations, for which a high inter-annotator agreement was reported [22, 7]. GRASS thus provides a good basis to model backchannel timing for specific turn-taking categories.

2.2. Data extraction

To build a model for predicting backchannel timing, it is essential to identify whether an utterance actually functions as a backchannel. This was achieved using the turn-taking annotations from the GRASS corpus, more specifically the "Points of Potential Completion" (PCOMP) tier. A PCOMP marks conversational turns based on syntactic completeness and conversational flow [7], focusing on subsequent conversational actions to minimize interpretative bias. PCOMPs are labeled for all continuously occurring utterances using a compact label set, including a label for utterances with a backchanneling function, referred to as Hearer Response Token (HRT) in GRASS. HRTs include typical backchannels such as "mhm", but may also consist of several tokens. For this paper, we extracted HRTs along with their preceding PCOMP labels and identified the corresponding portion of the partner's speech to analyze the context before each HRT. To do so, we traced back from the onset of the HRT through the partners PCOMP tier until reaching the immediately preceding turn, which we refer to as the pre-HRT (PHRT). If the PHRT overlapped with the HRT, only the portion occurring before the HRT onset was included, ensuring that overlapping speech is not incorporated into the dataset.

Figure 1 shows the relative distribution of turn-taking labels of the PHRTs for all extracted HRTs, with *hold* being the most common PCOMP label, accounting for approximately 50%. This is not surprising, as the *hold* label refers to those turns where speakers hold their turns and continue speaking. In the remainder of this paper, we only use PHRTs with the label *hold*, as other turn-taking functions did not occur frequently enough. Moreover, *hold* represents the most probable and natural context for a robot to produce a backchannel, making it the most suitable category for HRI. Based on these criteria, we extracted a total of 266 *hold*-HRT turn pairs for analysis.

3. Methods

3.1. Acoustic feature extraction

For the acoustic feature extraction, the last 1.025 seconds of each PHRT were segmented into overlapping 500 ms windows with a hop size of 25 ms. The extra 25 ms enabled the computation of differential features between consecutive windows. This process generated 20 segments per *hold* annotation, increasing the dataset size for training and allowing continuous prediction of backchannel timing via differential features. Figure 2 illustrates the windowing process and the extracted targets ($\Delta t_{i^{th}}$). In total, 5,123 segments remained after outlier removal. For each segment a total of 45 acoustic and prosodic features were extracted, of which 3 were durational features, 34 were related to F0 and RMS, and 8 to voice quality:

34 **Fundamental frequency (F0) and intensity (RMS) features**: These include measures such as mean, standard deviation, max/min values and their positions, and slope for each window, along with differential features capturing temporal changes. F0 extraction was performed using an autocorrelation algorithm from *parselmouth* [23], a Praat-based Python package, with speaker-specific F0 ranges: 60–300 Hz for males and 80–400 Hz for females. F0 values were extracted every 10 ms and smoothed using a moving average filter ($N = 5$) to reduce computational errors and ensure accurate max/min calculations. Intensity was extracted using *parselmouth* with default parameters, requiring no smoothing nor interpolation. To account for individual speaking style differences, all F0 and intensity features were speaker-normalized relative to the corresponding entire turn of each segment, as global normalization would not effectively capture natural variations within a conversation.

3 **Durational features**: As a correlate of articulation rate, we extracted the *voiced unvoiced ratio (VU-AR)*, because – unlike other measures for articulation rate – the computation of *VU-AR* does not require phonetic segmentation, word counts, or grapheme counts. The feature is computed as the zero-

crossing rate of a modified F0 representation (F0sign), where voiced frames are assigned +1 and unvoiced frames -1. The ratio is then derived from the normalized discrete difference of F0sign, thus providing a real-time alternative for estimating articulation rate. Further durational features were the *PHRT speaking duration*, i.e., the duration of the interlocutor's turn before the occurrence of the HRT, and the *PHRT HRT time gap*, which serves as the prediction target. Both were directly extracted from the PCOMP annotations.

8 **Voice quality features**: These features, all extracted using the *parselmouth* package, include harmonics-to-noise ratio (HNR) (via the `to_harmonicity` function), H1-H2 (computed using `to_pitch` and `to_spectrum` functions), as well as jitter and shimmer, extracted via the corresponding `call` functions. For each measure, the mean and the difference from the previous segment were computed.

3.2. Prediction

To identify the best model for predicting backchannel timing, i.e., the *PHRT HRT time gap*, we evaluated three ensemble learning models using the *PyCaret* Python library [24]. These models were chosen for their interpretability, compatibility with SHAP analysis, and support for regression tasks: (1) The light Gradient Boosting Machine (lightGBM) [25], which is a gradient boosting framework optimized for high-dimensional data and large datasets, (2) the Extra Trees Regressor (ET) [26], which is a randomized tree-based algorithm that averages multiple estimators to improve accuracy and reduce overfitting, and (3) the Random Forest Regressor (RF) [27], which is a robust and flexible ensemble learning algorithm that uses multiple decision trees.

Default parameter settings from *PyCaret* and the respective Python libraries were used for all models. Each model was trained with 100 estimators and unlimited tree depth. For lightGBM, a learning rate of 0.1 and up to 31 leaves per tree were used. For ET and RF, a minimum of two samples was required to split a node, and bootstrap sampling was enabled for RF but disabled for ET.

3.3. Evaluation

To evaluate the prediction models and ensure result reliability, we used standard metrics and applied ten-fold stratified *K-Fold* cross-validation with a 70/30 train-test split per fold. Root Mean Square Error (RMSE) was used to measure the difference between predicted and actual values while penalizing larger errors more heavily. The Mean Absolute Error (MAE) provides the average of absolute prediction errors in the same units as the target variable, offering an interpretable measure of typical error magnitude. The R^2 (Coefficient of Determination) indicates the proportion of variance in the target variable explained by the model, ranging from 0 (poor fit) to 1 (perfect fit) [28]. Additionally, we assessed computational efficiency using Prediction Time (PT). It assesses computational efficiency by averaging the time required for N individual predictions (see Equation 1), where time(y_i) is the time taken by the model to generate the i-th prediction of a randomly chosen sample y_i and N is the number of prediction samples used for averaging. The computations for the PT were performed on an Intel Xeon CPU E3-1230 v5 with 4 cores (8 threads), a clock speed of 3.40 GHz, and 32 GB RAM running on Windows 10, with $N = 1000$ individual predictions.

$$\text{PT} = \frac{\sum \text{time}(y_i)}{N} \quad (1)$$

Table 1: *Evaluation metrics for predicting PHRT HRT time gap using only the 10 most important features for lightGBM, ET and RF.*

Model	MAE	RMSE	R^2	PT
lightGBM	0.1335	0.1955	0.6391	1.48 ms
ET	0.1016	0.1948	0.6398	27.08 ms
RF	0.1179	0.1917	0.6527	26.76 ms

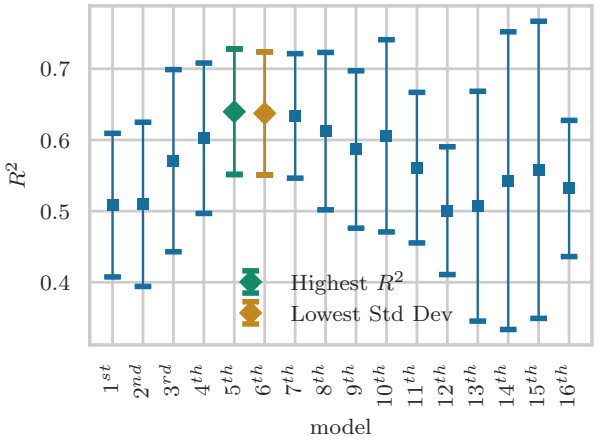

Figure 3: *Mean R^2 values with standard deviations for each 5-window combination in the lightGBM model. The best-performing model is highlighted in green, while the model with the lowest standard deviation is highlighted in yellow.*

To gain insights into feature impact on prediction outcomes, we used two interpretability methods: (1) Permutation Feature Importance, which assesses a feature's relevance by shuffling its values and measuring the effect on the model's score [29], and (2) SHAP values, which were originally derived from cooperative game theory. Unlike permutation feature importance, which provides a global ranking of features, SHAP values offer a more granular view, quantifying how each feature individually contributes to a specific prediction [30].

4. Results and discussion

4.1. Model performance

Table 1 presents the evaluation metrics for each model. The lightGBM model demonstrated a clear advantage in efficiency, being nearly 15 times faster in prediction time compared to the other models. Despite the Random Forest Regressor achieving the lowest RMSE and highest R^2–making it the most accurate overall–and the Extra Trees Regressor having the lowest MAE, all three models performed similarly in terms of RMSE and R^2. Given its strong performance and computational efficiency, lightGBM emerges as the most suitable choice for real-time applications. It achieved an MAE of approximately 130 ms, which falls within the expected variation of turn transition durations (e.g., [31]) and is faster than typical speech production reaction times [32]. Additionally, its R^2 value of 0.64 indicates that it captures a substantial portion of variance in the data.

As a next evaluation step, we searched for the speech segment that provides the most accurate prediction of the *PHRT HRT time gap*. To achieve this, we trained 16 separate instances

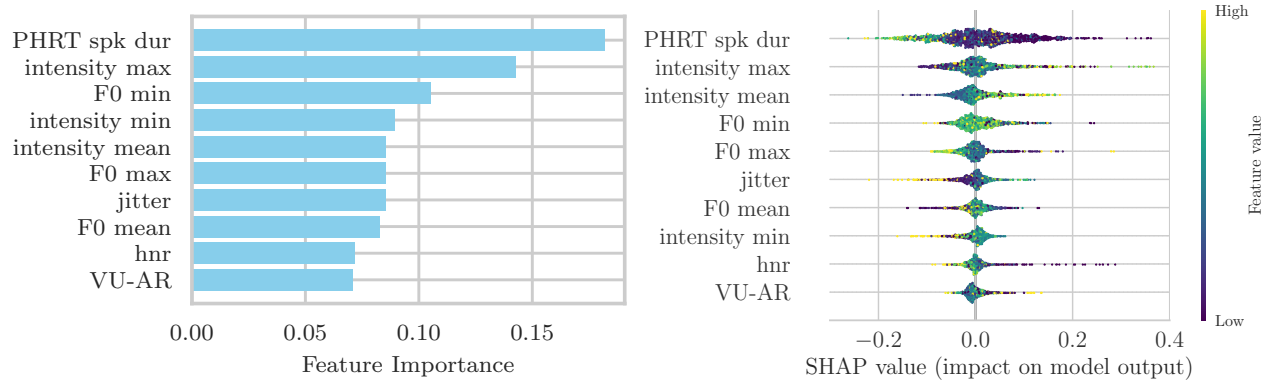

Figure 4: *Feature importance plot showing the 10 most important features based on permutation feature importance (left) and a beeswarm plot of the same features ranked by mean SHAP values (right), using the lightGBM model trained on the full dataset. Features are ordered hierarchically, with the most impactful ones at the top.*

of the lightGBM model, each using a different 5-window combination of the input data (e.g., windows 1–5, 2–6, ..., 16–20), resulting in 16 models, each with 1256 data points. Between 3.4% and 5.4% of data points had missing values, which were mean-imputed before training.

Figure 3 displays the R^2 values for all window combinations. The 5^{th} model, corresponding to windows 5-9 (125–725 ms), achieved the highest R^2 value. This corresponds to a speech segment occuring 275–875 ms before an HRT. This result aligns with the expectation that prediction performance decreases as the prediction window is farther away from an HRT, and also decreases as the prediction window approaches the start of an HRT or in other words, the end of an utterance.

4.2. Feature importance analysis

Feature importance analysis using the lightGBM model was conducted through the two previously introduced evaluation methods. Both methods identified *PHRT speaking duration* and maximum intensity as the top-ranked features in predicting the *PHRT HRT time gap* (see Figure 4). Beyond these top two features, those ranked 3–10 showed only minor differences in importance.

PHRT speaking duration was identified as the most important feature, with longer speaking durations leading to shorter *PHRT HRT time gaps*, while shorter durations resulted in longer gaps (see Figure 4, right). Additionally, *VU-AR*, another durational measure, ranked among the top ten features, further reinforcing the importance of temporal aspects in backchannel timing. This aligns with previous studies emphasizing the role of temporal features in signaling turn-taking cues (e.g., [33], [34]).

Intensity features, particularly maximum intensity, proved to be stronger predictors in our experiments than previously reported [35]. This may be due to our feature extraction method, which continuously tracks dynamic changes in loudness. This aligns with findings by Inoue et al. [17], who demonstrated that deleting prosodic information related to intensity, such as flattening the intensity curve, decreased the overall performance of their backchannel prediction model. Their results suggest that intensity plays a crucial role in predicting backchannel timing, a conclusion reinforced by our own findings, where intensity emerged as one of the most influential features.

While F0 features ranked among the top, the expected link between flat or falling F0 contours and backchannels [13] was not conclusively observed, suggesting limitations in the current modeling approach. Similarly, Inoue et al. [17] observed that flattening the F0 curve led to a small negative impact on their model's performance, although this effect was less pronounced than the impact observed when removing intensity-related prosodic information.

Finally, two voice quality features, jitter and HNR, were also present among the top-ranked predictors. Although their overall contribution to the model was less pronounced compared to duration and intensity, they still appear to play a secondary role in signaling backchannel opportunities, which was also shown by [34]. Interestingly, the SHAP values for HNR showed a relatively clear trend, where low HNR values correlated with increased prediction output.

5. Conclusion

This study analyzed the impact of prosodic features on backchannel timing and proposed methods for its prediction for HRI. The used model, selected for its efficiency and strong performance, achieved a mean average prediction error of approx. 130 ms, using the 10 most important features. This error margin falls within the expected variation of turn transition durations (e.g., [31]) and is faster than average reaction times for speech [32]. Feature importance analysis revealed that speaking duration, and intensity are among the most important predictors of backchannel timing. These results demonstrate the feasibility of using prosodic features and machine learning for continuous real-time backchannel prediction in conversational agents.

While the results are promising, the model has limitations. Notably, it relies exclusively on prosodic features and excludes visual and contextual cues [6]. Incorporating multi-modal features, such as gaze or facial expressions, could enhance the robustness and accuracy of future backchannel predictions. Additionally, the model should be tested in an HRI scenario (e.g., with Furhat) to assess whether the generated backchanneling behavior enhances interaction quality and user engagement.

To conclude, our findings provide a strong foundation for developing more advanced backchanneling models and conversational agents. By exploring the prosodic cues that influence backchannel timing, the study offers valuable insights for improving the naturalness of HRI. Ultimately, this research contributes to the broader goal of advancing HRI technology and enhancing the naturalness of conversational agents.

6. Acknowledgements

This research was funded in part by the Austrian Science Fund (FWF) [10.55776/P32700].

7. References

[1] T. Belpaeme, J. Kennedy, A. Ramachandran, B. Scassellati, and F. Tanaka, "Social robots for education: A review," *Science robotics*, vol. 3, no. 21, p. eaat5954, 2018.

[2] C. A. Cifuentes, M. J. Pinto, N. Céspedes, and M. Múnera, "Social robots in therapy and care," *Current Robotics Reports*, vol. 1, pp. 59–74, 2020.

[3] O. Mubin, M. I. Ahmad, S. Kaur, W. Shi, and A. Khan, "Social robots in public spaces: A meta-review," in *Proc. of ICSR*. Springer, 2018, pp. 213–220.

[4] Štefan Beňuš, A. Gravano, and J. Hirschberg, "Pragmatic aspects of temporal accommodation in turn-taking," *Journal of Pragmatics*, vol. 43, no. 12, pp. 3001–3027, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378216611001469

[5] E. A. Schegloff, *Discourse as an interactional achievement: Some uses of "uh huh" and other things that come between sentences*. Analyzing discourse: Text and talk/Georgetown University Press, 1982.

[6] S. Bögels and F. Torreira, "Turn-end estimation in conversational turn-taking: The roles of context and prosody," *Discourse Processes*, vol. 58, no. 10, pp. 903–924, 2021.

[7] A. Kelterer and B. Schuppler, "Turn-taking annotation for quantitative and qualitative analyses of conversation," 2025. [Online]. Available: https://arxiv.org/abs/2504.09980

[8] R. Ogden, "The phonetics of talk in interaction – introduction to the special issue," *Language and Speech*, vol. 55, no. 1, pp. 3–11, 2012, pMID: 22480023. [Online]. Available: https://doi.org/10.1177/0023830911433559

[9] S. Al Moubayed, J. Beskow, G. Skantze, and B. Granström, "Furhat: A back-projected human-like robot head for multiparty human-machine interaction," in *Cognitive Behavioural Systems*. Springer Berlin Heidelberg, 2012, pp. 114–130.

[10] G. Skantze, A. Hjalmarsson, and C. Oertel, "Turn-taking, feedback and joint attention in situated human–robot interaction," *Speech Communication*, vol. 65, pp. 50–66, 2014.

[11] M. Johansson, T. Hori, G. Skantze, A. Höthker, and J. Gustafson, "Making turn-taking decisions for an active listening robot for memory training," in *Social Robotics*. Springer International Publishing, 2016, pp. 940–949.

[12] D. Lala, P. Milhorat, K. Inoue, M. Ishida, K. Takanashi, and T. Kawahara, "Attentive listening system with backchanneling, response generation and flexible turn-taking," in *Proc. of SIGDial*. Association for Computational Linguistics, 2017.

[13] N. Ward and W. Tsukahara, "Prosodic features which cue back-channel responses in English and Japanese," *Journal of Pragmatics*, vol. 32, no. 8, pp. 1177–1207, 2000.

[14] J. Y. Jang, S. Kim, M. Jung, S. Shin, and G. Gweon, "Bpm_mt: Enhanced backchannel prediction model using multi-task learning," in *Proc. of EMNLP*, 2021, pp. 3447–3452.

[15] W. Liermann, Y.-H. Park, Y.-S. Choi, and K. Lee, "Dialogue act-aided backchannel prediction using multi-task learning," in *Proc. of Findings: EMNLP*, 2023, pp. 15 073–15 079.

[16] E. Ekstedt and G. Skantze, "Voice activity projection: Self-supervised learning of turn-taking events," in *Interspeech 2022*, 2022, pp. 5190–5194.

[17] K. Inoue, D. Lala, G. Skantze, and T. Kawahara, "Yeah, un, oh: Continuous and real-time backchannel prediction with fine-tuning of voice activity projection," in *Proc of NAACL: Human Language Technologies*. Albuquerque, New Mexico: Association for Computational Linguistics, 2025, pp. 7171–7181. [Online]. Available: https://aclanthology.org/2025.naacl-long.367/

[18] R. Poppe, K. P. Truong, and D. Heylen, "Backchannels: Quantity, type and timing matters," in *Intelligent Virtual Agents*. Springer Berlin Heidelberg, 2011, pp. 228–239.

[19] Paierl, Michael, "Modeling backchannels for human-robot interaction," Master's thesis, 2024.

[20] B. Schuppler, M. Hagmüller, J. A. Morales-Cordovilla, and H. Pessentheiner, "GRASS: The Graz corpus of Read And Spontaneous Speech," in *Proc. of LREC*, 2014, pp. 1465–1470.

[21] B. Schuppler, M. Hagmüller, and A. Zahrer, "A corpus of read and conversational Austrian German," *Speech Communication*, vol. 94, 09 2017.

[22] B. Schuppler and A. Kelterer, "Developing an annotation system for communicative functions for a cross-layer ASR system," in *Proc. of DiscAnn*. Association for Computational Linguistics, 2021, pp. 14–18. [Online]. Available: https://aclanthology.org/2021.discann-1.3

[23] Y. Jadoul, B. Thompson, and B. de Boer, "Introducing parselmouth: A python interface to praat," *Journal of Phonetics*, vol. 71, pp. 1–15, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0095447017301389

[24] M. Ali, *PyCaret: An open source, low-code machine learning library in Python*, April 2020, pyCaret version 1.0.0. [Online]. Available: https://www.pycaret.org

[25] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Proc. of NIPS*, vol. 30, pp. 3146–3154, 2017.

[26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *JMLR*, vol. 12, pp. 2825–2830, 2011.

[27] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995, pp. 278–282.

[28] C. Hagquist and M. Stenbeck, "Goodness of fit in regression analysis–R^2 and G^2 reconsidered," *Quality and Quantity*, vol. 32, no. 3, pp. 229–245, 1998.

[29] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[30] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. of NIPS*. Curran Associates, Inc., 2017, pp. 4765–4774. [Online]. Available: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

[31] M. Heldner and J. Edlund, "Pauses, gaps and overlaps in conversations," *Journal of Phonetics*, vol. 38, no. 4, pp. 555–568, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0095447010000628

[32] D. Fry, "Simple reaction-times to speech and non-speech stimuli," *Cortex*, vol. 11, no. 4, pp. 355–360, 1975. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S001094527580027X

[33] F. Torreira and S. Bögels, "Vocal reaction times to speech offsets: Implications for processing models of conversational turn-taking," *Journal of Phonetics*, vol. 94, p. 101175, sep 2022.

[34] G. Skantze, "Turn-taking in conversational systems and human-robot interaction: a review," *Computer Speech & Language*, vol. 67, p. 101178, 2021.

[35] A. Gravano and J. Hirschberg, "Turn-taking cues in task-oriented dialogue," *Computer Speech & Language*, vol. 25, no. 3, pp. 601–634, 2011.