Predicting Errors and Failures in Human-Robot Interaction from Multi-Modal Temporal Data

Ruben Janssens
ruben.janssens@ugent.be
IDLab-AIRO, Ghent University - imec
Ghent, Belgium

Eva Verhelst
eva.verhelst@ugent.be
IDLab-AIRO, Ghent University - imec
Ghent, Belgium

Mathieu De Coster
mathieu.decoster@ugent.be
IDLab-AIRO, Ghent University - imec
Ghent, Belgium

Abstract

Social robots should be able to detect social signals sent by their user, such as when the robot made a mistake or the user feels awkward. As our submission to the ERR@HRI challenge, we present a number of neural and traditional machine learning models to predict when this occurs in a human-robot conversation, based on facial expressions, body pose, and non-verbal speech characteristics. The small size of the dataset, imbalance in the label distribution, and low-grained label annotations provided significant challenges. However, three of our approaches show promising results: modifying the training of a gated recurrent unit (GRU) model to predict at lower frequency than that of the input features, using an embedding layer and convolutional neural network to pre-process temporal data before feeding it to the GRU, and using traditional random forests.

CCS Concepts

• **Human-centered computing** → **Human computer interaction (HCI)**; • **Computer systems organization** → **Robotics**; • **Computing methodologies** → **Machine learning**.

Keywords

social signal processing, human-robot interaction, multimodal machine learning

ACM Reference Format:
Ruben Janssens, Eva Verhelst, and Mathieu De Coster. 2024. Predicting Errors and Failures in Human-Robot Interaction from Multi-Modal Temporal Data. In *INTERNATIONAL CONFERENCE ON MULTIMODAL INTERACTION (ICMI '24), November 4–8, 2024, San Jose, Costa Rica.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3678957.3688388

1 Introduction and Related Work

Many applications of human-robot interaction (HRI) stand to benefit from social robots that can be more aware of social signals that are sent by a user in a human-robot conversations. Take an educational social robot, which should be able to adapt the interaction when it detects that the user does not understand what it just said [1]. However, automatically processing such social signals in HRI has been shown to be a very challenging task, that is highly context-dependent [3]. It is also a highly multi-modal endeavour, seen as a promising application of integrating the multiple modalities available in HRI [4]. Earlier work has shown that both neural networks such as Bidirectional LSTMs as well as traditional machine learning models such as Random Forests are promising for detecting social signals such as user uncertainty from data containing multiple modalities such as facial expressions and body pose [2].

In particular, detecting errors and failures is an important aspect of processing social signals in human-robot interaction. As such, the goals of the ERR@HRI challenge is to automatically detect moments of robot mistakes, user awkwardness, or interaction ruptures (defined as either a robot mistake or user awkwardness), in human-robot conversations. The data provided is collected from videos of interactions of employees with a coaching robot [5, 6]. It consists of features extracted from three modalities present in the data: facial expressions, represented as Facial Action Units (FAUs) detected by the toolkit OpenFace, body pose, detected by OpenPose, and features representing non-verbal parameters of speech, extracted by OpenSMILE. Ground truth labels are provided separately for the three tasks (Robot Mistake, User Awkwardness, and Interaction Ruptures), each of which are seen as a binary classification task, with label "1" representing that the phenomenon is present.

In this paper, we explore the use of different machine learning approaches for this task, evaluating how to optimise these models for this data set. From these approaches, we selected three models per task to submit to the challenge, and we report their performance on the unseen test data.

2 Models

We explored a number of model architectures, starting from the baseline models and using insights from the data as described in the previous section. This section describes these architectures and the various optimisation approaches we have evaluated.

All models used the dataset with features normalised to zero mean and unit variance, with the distributions learned on the training set. Unless otherwise noted, the data is preprocessed as in the code provided by the challenge organisers, providing for each time step (at 30 frames per second) a sequence containing the input features of the last n time steps, n being referred to as the "sequence length" in this paper.

Each model's performance is evaluated on the validation data set. We used the metrics as provided by the challenge organisers: accuracy, F1 score, precision, recall, as well as these metrics with one timeframe of tolerance applied to them. Given that the "tolerant" scores were very close to the non-tolerant scores, we focused on the

ICMI '24, November 4–8, 2024, San Jose, Costa Rica

ACM ISBN 979-8-4007-0462-8/24/11
https://doi.org/10.1145/3678957.3688388

scores without tolerance. All models were trained for 100 epochs, but early stopping is used to select the epoch with the highest validation metric value, unless otherwise indicated.

As the F1 score calculated by the challenge-provided code corresponds to the "macro" F1 score, the average of the F1 scores for both 0 and 1 as positive labels, we also calculated the "micro" F1 score, where only 1 is seen as the positive label – this because the macro F1 score does not sufficiently penalise a model that disproportionally often predicts the majority label. However, our focus remains on the accuracy and macro F1 scores, as those were the objectives for the challenge.

2.1 Baseline models: GRU and BiLSTM

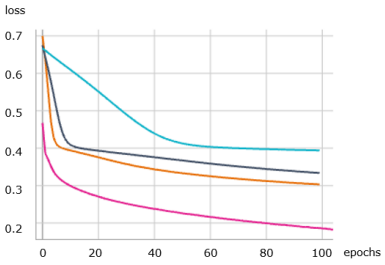
As the challenge organisers provided a baseline model for each of the subtasks, we started by recreating these architectures. They use either one gated recurrent unit (GRU) layer or one bi-directional Long Short-Term Memory (BiLSTM) layer, both with 128 hidden features, and a drop-out layer with probability 0.2, and a linear layer to reduce the hidden layer to one scalar value, on which a sigmoid activation function is applied, as this is a classification task.

We investigated the effect of a number of hyperparameter settings of these models. First of all, the models' training curves showed that they overfitted very quickly. To resolve this issue, we changed the learning rate from the default $1e^{-4}$ to $1e^{-5}$, $5e^{-6}$, and $1e^{-6}$. While the highest scores reached during training for each of these learning rates do not differ significantly (for the GRU model on the robot mistake task: accuracy = 0.8536 and macro F1 = 0.5478 for learning rate $1e^{-4}$ vs. accuracy = 0.8538 and macro F1 = 0.5528 for learning rate $1e^{-5}$, which performed best of all learning rates tested), the training curves show slower overfitting, so we use learning rate $1e^{-5}$ in the next experiments.

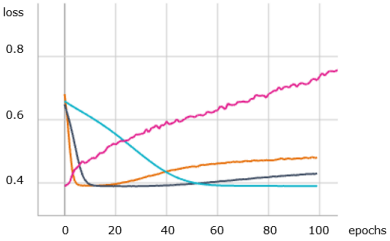
Data exploration showed that all three target phenomena occur over relatively long timespans, with continuous stretches of label "1" lasting, on average, 9.5s for Robot Mistake and 15.0s for User Awkwardness, corresponding to 285 and 450 frames respectively. This is much longer than the sequence length of 5 frames as used in the baseline models. We tested sequence lengths 20, 40, and 60 for the GRU for Robot Mistake. Performance barely changed: sequence length 40 with learning rate $1e^{-5}$ performed best, with accuracy = 0.8504 and macro F1 = 0.5472, slightly lower than with sequence length 5. No higher sequence lengths than 60 were tested, as this already required 45GB of CPU memory in our implementation.

However, looking at the actual predictions of the model on sample data shows that the model predicts label "0" disproportionately often. The dataset is highly imbalanced: for both Robot Mistake and User Awkwardness, only 16% of the time in training and validation sessions contains label "1", for Interaction Rupture, this is 26%. To better quantify how well the model handles this unbalanced data, we also calculated the "micro" F1 score, for "1" as positive label. The GRU model for Robot Mistake, with learning rate $1e^{-5}$, reaches a maximum micro F1 score of 0.092 (compared to macro F1 = 0.5528), indicating that the model did not really learn patterns except for the dominance of the label "0".

In order to combat this, we tested the impact of using a weighted loss function, weighing the loss for the positive class with the proportion of negative over positive labels. For the GRU, with learning rate $1e^{-5}$ and sequence length 5, this results in a micro F1 score of 0.3014 instead of 0.092 without weighted loss. This micro F1 score is already achieved after one epoch of training. At that point, the macro F1 score is 0.4088 and the accuracy 0.4525, much lower than without a weighted loss. They achieve a maximum of macro F1 = 0.4914 and accuracy = 0.6373, both after 100 epochs of training, still indicating that even though the model seems to learn more detailed patterns, it is still not enough to surpass the models that predict "0" for almost all frames at accuracy and macro F1.



(a) Training loss.



(b) Validation loss.

Figure 1: Training curves showing loss of the baseline GRU model after each epoch of training, for the Robot Mistake task, using learning rate $1e^{-4}$ (in pink), $1e^{-5}$ (in orange), $5e^{-6}$ (in dark grey), and $1e^{-6}$ (in cyan).

Using a weighted loss, higher sequence lengths than 5 showed a slightly positive effect on scores, although still very small: sequence length 20 achieved a micro F1 score of 0.3082 after two epochs, and macro F1 of 0.5118 and accuracy of 0.6855 (after 96 epochs).

Finally, as another attempt to reduce overfitting, we increased the dropout probability from 0.20 to 0.40. This allowed us to achieve a slightly higher micro F1 score of 0.3086 using the model with sequence length 40 (compared to 0.3012 with dropout probability 0.20), however it reduced the maximum accuracy from 0.6731 to 0.653 and maximum macro F1 score from 0.5057 to 0.4989. For sequence length 20, the impact was (very slightly) negative on the micro F1 score as well, going from 0.3082 to 0.3049.

2.2 Dense layers

As the baseline models quickly overfitted on the training data, we also established a simpler baseline, using simple dense layers. As a dense layer is not able to process temporal information, we provided it with the average of the input features over the five frames in the input sequence. We tested (i) a single linear layer with 128 hidden features, (ii) two such layers with a ReLU activation after the first

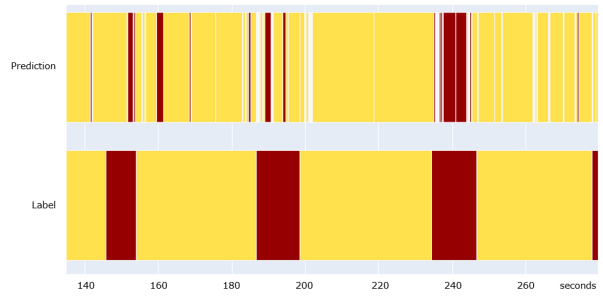


Figure 2: Ground truth labels (yellow is '0', red is '1') and predictions of the baseline GRU model after 10 epochs of training for the User Awkwardness task, showing an excerpt of a session from the validation set, with the x-axis showing number of seconds since start of the session.

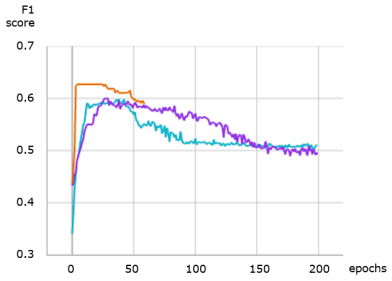
layer, (iii) these two layers but using only one frame of context instead of the average of the last five frames and (iv) the same two layers but with 512 hidden features. Using different learning rates, but not using a weighted loss, most of these models reach a plateau that none of them can exceed. For Robot Mistake, this plateau lies at accuracy = 0.856 and macro F1 score of 0.5819, for User Awkwardness: accuracy = 0.8285 and macro F1 = 0.6269, for Interaction Rupture: accuracy = 0.768 and macro F1 = 0.5416.

2.3 Coarse-grained predictions

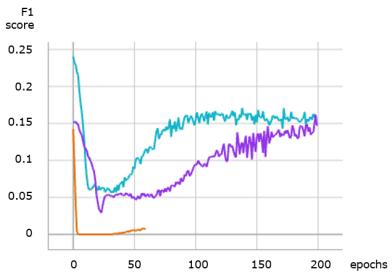
Visualising the predictions of the baseline GRU models, as can be seen in Figure 2, shows that when the models predict the label "1", this is often predicted in contiguous sequences of only a few frames long. However, in the ground truth, these contiguous sequences are often hundreds of frames long, as mentioned above. This could indicate that the model tries to learn patterns that are more fine-grained than actually present in the training data. In an attempt to mitigate this, we tried two methods to get the models to make predictions that are more coarse-grained.

First, we used the existing baseline model (here: the GRU trained for 10 epochs for the User Awkwardness task, all other settings the same as the original baseline), but applied a "low-pass filter" on the predictions. This filter was implemented by applying a sliding window over the model's predictions and taking the median of all predictions in that window. Testing window sizes from 10 to 190 frames in steps of 10, the highest accuracy was achieved with a window size of 140, reaching accuracy = 0.7868 compared to 0.7665 without low-pass filter. Macro F1 score did not improve using a low-pass filter, as recall was consistently lower, but precision did slightly improve from 0.5660 to 0.5709 using a window size of 40. Overall, it seems the low-pass filter has a slight potential benefit, but not sufficient to be included in subsequent approaches.

Second, we re-trained the baseline models, modifying the training and prediction process to predict only one single value per sequence – we call this a "coarse-grained" model. For example, for a sequence length of five, the original models predicted a value for each frame, given the last five frames of input features as context. The coarse-grained models partition the training data into



(a) Macro F1 score.



(b) Micro F1 score.

Figure 3: Training curves showing validation scores of the coarse-grained model after each epoch of training, for the User Awkwardness task, using learning rate $1e^{-5}$ and sequence lengths 5 (in orange), 20 (in cyan), and 40 (in purple), training for 200 epochs.

non-overlapping sequences of five frames, for which the model predicts only one value. The majority label in each sequence is used as ground truth label during training, while during validation, the original granularity is maintained to ensure comparability with other models.

Testing this coarse-grained approach with learning rates $1e^{-5}$ and $5e^{-5}$ as well as sequence lengths 5, 20, and 40, and using an unweighted loss, we see again that the models are again not able to surpass the accuracy and macro F1 scores reported for the dense layer baselines. However, looking at the micro F1 score training curves of the coarse-grained models, shown in Figure 3, shows that when the macro F1 score is highest, the micro F1 score is 0, indicating that the models predict 0 for all training instances.

However, the models also do seem to learn more detailed patterns, as the micro F1 score reaches a value higher than 0 both before and after the plateau in which it stays 0 for a number of epochs. For User Awkwardness, the model with sequence length 20 reaches a micro F1 score of 0.2405, for Robot Mistake, it reaches 0.2655 (both after one epoch), and for Interaction Rupture, 0.3355, with sequence length 40 and after four epochs. As the training curves show, for the micro F1 score, a longer sequence length seems to lead to a higher score, but also requiring more epochs of training. Throughout the training curves, it is clear that there is an inverse correlation between the micro F1 score on one hand and the macro F1 score and accuracy on the other hand.

2.4 CNN with embedding layer

Building on the lessons learned in the previous sections, we investigate one more approach to teach the model to process the data in a more coarse-grained manner. Convolution Neural Networks (CNNs), while most-often used for spatial data such as images and video, can also be used for temporal data, as shown by Van Den Oord et al. for audio streams [7]. In this case, the convolutional kernels downsample the original input data streams to a lower frequency, recognising patterns in the process.

In addition, as the dataset is relatively high-dimensional with 90 input features, and given that some correlations exist between some of the features (e.g. FAUs being represented by two features, one describing the presence of the FAU and one describing the intensity), we investigate using an embedding layer. This layer maps the input feature space to a lower-dimensional space, aiming to make more efficient use of the subsequent layers of the model.

This results in the following architectures: (i) a linear (embedding) layer with ReLU activation function, followed by a GRU in order to process the temporal dimension, with dropout and linear classification layer as in the baseline models, (ii) a one-dimensional CNN layer with ReLU activation, again followed by a GRU layer with dropout and linear layer, and (iii) first the linear embedding layer, followed by the CNN layer, and then again the GRU and final linear layer.

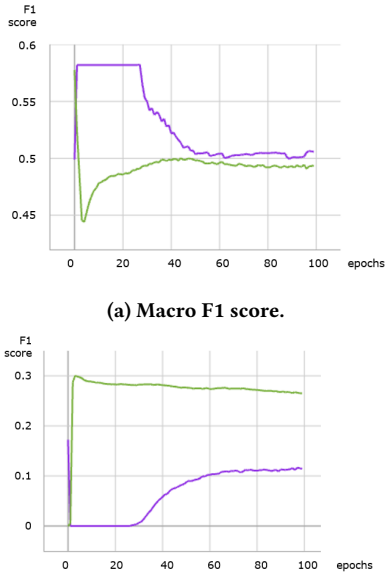
For the first (embedding-GRU) model, we use an embedding size of 32 features, slightly bigger than the GRU size of 24 hidden features. For all models, no weighted loss is used, sequence length is 20 and learning rate $1e^{-5}$, and performance on the Robot Mistake task is reported, unless otherwise mentioned. This model reaches a maximum accuracy of 0.6483 and macro F1 score of 0.4992, after 99 epochs, with a micro F1 score of 0.2677. The maximum micro F1 score is reached after 5 epochs, and is 0.2848. Interestingly, the micro F1 score does not reach very low values, indicating this approach is not prone to learning all-zero predictions, even without using a weighted loss.

For the second (CNN-GRU) model, we use a CNN kernel size of 4 and stride of 2, and compare using 16, 24 or 32 filters. 24 filters leads to the best performance, with a maximum accuracy of 0.6626 and macro F1 score of 0.5129 after just one epoch, and micro F1 score of 0.2568. Maximum micro F1 score of 0.302 is reached after three epochs, with accuracy of 0.5248 and macro F1 score of 0.4525.

Finally, the embedding-CNN-GRU model initially predicts all-zero labels, but, as shown in the training curves in Figure 4, the micro F1 score again increases after ca. 30 epochs. Maximum micro F1 score is after one epoch, reaching 0.1725, with accuracy = 0.7241 and macro F1 = 0.499. After 100 epochs, micro F1 score reached 0.1142, with accuracy = 0.8016 and macro F1 = 0.5058. When using a weighted loss, as also shown in Figure 4, the model starts with all-zero predictions after one epoch but reaches a maximal micro F1 score of 0.3003 after four epochs, with accuracy = 0.5129 and macro F1 = 0.4456.

2.5 Random Forest

Finally, we also explored the performance of Random Forests on this task, a non-neural machine learning model which is often-used and robust for many tasks with smaller datasets, especially tabular data.



(a) Macro F1 score.

(b) Micro F1 score.

Figure 4: Training curves showing validation scores of the embedding-CNN-GRU model after each epoch of training, without (in purple) and with (in green) weighted loss.

Table 1: Validation performance of the best-performing Random Forest.

	Accuracy	Macro F1	Micro F1
Robot Mistake	0.8474	0.4953	0.0738
User Awkwardness	0.8056	0.5212	0.1522
Interaction Rupture	0.7294	0.5605	0.2880

A random forest has relatively few settings: the most important are the number of decision trees that constitute the model, the amount of training samples used for fitting each tree, and the minimum amount of samples in a leaf of each tree. We investigated building forests consisting of 10 up to 101 trees, using 50,000 to 200,000 samples per tree, and 5 to 15 as minimum amount of samples per leaf. These tests showed that an odd number of trees leads to a higher performance, likely because an even number of trees could predict "0.5" as output value in case of a tie. Furthermore, using a higher minimum amount of samples per leaf reduced overfitting. Using balanced class weights was also tested, but did not provide a benefit to the performance of the model.

We also evaluated multiple approaches to providing the temporal context to the random forest. As simple approaches, we tried only providing the last frame, only providing the average of each input feature over the sequence length, and providing all of the input features over the sequence length. Finally, we tested a more engineered approach, providing the first, middle, and last input values of the sequence, as well as the mean value over the entire sequence and the mean over the latter half of the sequence.

Table 2: Results of submitted models for test data, comparing with organiser baseline. Best models indicated with *.

Task	Model	Accuracy	Precision	Recall	Macro F1	Tolerant acc.	Tolerant prec.	Tolerant recall	Tolerant F1 score
Robot Mistake	Coarse-grained (20 eps)	0.82	0.50	0.48	0.46	0.82	0.50	0.48	0.46
	GRU (weighted loss)	0.59	0.58	0.55	0.52	0.60	0.58	0.55	0.52
	* Embedding-CNN-GRU	0.80	0.56	0.53	0.52	0.81	0.59	0.54	0.53
	Baseline	0.71	0.56	0.54	0.54	0.71	0.56	0.54	0.54
User Awkwardness	* Coarse-grained (40 eps)	0.77	0.67	0.53	0.50	0.77	0.68	0.53	0.50
	Coarse-grained (190 eps)	0.72	0.55	0.53	0.53	0.72	0.56	0.54	0.54
	Random Forest	0.73	0.52	0.51	0.49	0.75	0.58	0.53	0.51
	Baseline	0.73	0.56	0.57	0.57	0.73	0.57	0.58	0.57
Interaction Rupture	Coarse-grained (20 eps)	0.70	0.68	0.54	0.49	0.70	0.69	0.54	0.50
	Coarse-grained (190 eps)	0.66	0.57	0.55	0.55	0.67	0.58	0.56	0.56
	* Random Forest	0.69	0.61	0.55	0.54	0.72	0.68	0.59	0.58
	Baseline	0.68	0.56	0.50	0.42	0.69	0.59	0.50	0.42

The best-performing model consisted of 19 decision trees, using a sequence length of 40 and the engineered approach described above, with 200,000 samples per tree and minimum 10 samples per leaf. The performance of this model is shown in Table 1.

3 Results

In our submission, we provided three models for each task, in order to evaluate the merits of the different approaches we looked at in this work. For each of the three tasks, we submitted a coarse-grained model with sequence length 20, that was trained for only 20 epochs (for Robot Mistake and Interaction Rupture) or 40 epochs (for User Awkwardness), as these models reached the highest accuracy and macro F1 score – however, these models predicted almost only "0" labels.

For the User Awkwardness and Interaction Rupture task, we also submitted the coarse-grained model with sequence length 20 after 190 epochs of training, as these reached the highest micro F1 score (barring after their first epoch of training). We also submitted the Random Forest with settings described in the previous section, as this model was able to achieve a relatively high accuracy, micro F1 and macro F1 score at the same time, especially for the Interaction Rupture task.

Finally, for the Robot Mistake task, we submitted the GRU model with weighted loss, sequence length 40, and dropout 0.40, after 10 epochs of training, as this model achieved the highest micro F1 score there. Last, we submitted the embedding-CNN-GRU model with sequence length 20 and 16 CNN filters, after 90 epochs of training without weighted loss, as this model provided the best balance between high micro F1 score and high accuracy and macro F1 score. Both of these approaches were only tested for the Robot Mistake task.

Table 2 shows the results of these nine submitted models on the test data. Interestingly, for each subtask, a different approach was the overall best (counting all metrics together).

The GRU with weighted loss performed worst for of Robot Mistake models, indicating that purely optimising the micro F1 score was not effective for these metrics. This same conclusion can be drawn from the User Awkwardness results. For the Robot Mistake, the embedding-CNN-GRU model seems to have struck a good balance between a high micro F1 score and high accuracy and macro F1 score, that generalised well to the test data.

For Interaction Rupture, the Random Forest performs best. On the validation data, the Random Forest also performed much better for Interaction Rupture than for the other two tasks, suggesting that either the underlying structure of that data is different from the other task, somehow being more suitable for Random Forests, or that the model settings should be tuned differently for the other two tasks.

4 Conclusion and Future Work

As our submission to the ERR@HRI challenge, we developed a number of different neural and traditional machine learning models to predict robot mistakes, user awkwardness and interaction ruptures from data representing facial expressions, body pose, and non-verbal speech parameters.

The development of these models was challenging, as neural models tended to easily overfit, and the input data was much more fine-grained than the ground truth label annotation. The former problem was mitigated by modifying learning rate and dropout parameters, while for the latter, we tested various approaches to process the temporal dimension of the data while keeping the balance with the coarse-grained ground truth. These approaches included reducing the frequency of the output labels during training time and using an embedding and CNN layer to pre-process the temporal dimension before the GRU layer, the latter of which showed to be especially promising for the Robot Mistake task.

In addition, the unbalanced nature of the dataset proved to be challenging, as it required a careful choice of metrics to optimise for. Besides accuracy and macro F1 score, as used by the challenge, we chose to also optimise for the micro F1 score, in order to investigate whether our models could learn more detailed patterns, as optimising for the macro F1 score and accuracy often resulted in models only predicting the dominant class. Results showed that optimising for both the macro and micro F1 score led to better-performing

models in the challenge leaderboard than only optimising for either macro F1 score or micro F1 score.

Finally, the results also indicated that non-neural machine learning models, namely random forests, remain powerful tools for smaller datasets like these.

Future work should focus on investigating the importance of the different modalities and individual features, as well as more detailed analysis of the fragments of interactions where the models make mistakes. Our work did not focus on these aspects, as this is more difficult without the original video and audio data. Other interesting approaches could include leveraging transfer learning from other affect recognition tasks, or perhaps using pre-trained embeddings for the different modalities.

Detecting mistakes or awkwardness in human-robot conversations remains a challenging but highly important task for human-robot interaction. We hope that this work contributes to this research field and look forward to future progress in this area.

Acknowledgments

This research was funded in part by the Flemish Government (AI Research Program) and imec's Smart Education program. We thank our colleagues Giulio Antonio Abbo, Maria Jose Pinto Bernal, Qiaoqiao Ren, and our supervisors Tony Belpaeme and Thomas Demeester for their input during discussions about this challenge.

References

[1] Tony Belpaeme, James Kennedy, Aditi Ramachandran, Brian Scassellati, and Fumihide Tanaka. 2018. Social robots for education: A review. *Science robotics* 3, 21 (2018), eaat5954.

[2] Ronald Cumbal, José Lopes, and Olov Engwall. 2020. Detection of listener uncertainty in robot-led second language conversation practice. In *Proceedings of the 2020 International Conference on Multimodal Interaction*. 625–629.

[3] Hatice Gunes and Nikhil Churamani. 2023. Affective Computing for Human-Robot Interaction Research: Four Critical Lessons for the Hitchhiker. In *2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 1565–1572.

[4] Ruben Janssens. 2024. Multi-modal Language Models for Human-Robot Interaction. In *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*. 109–111.

[5] Micol Spitale, Minja Axelsson, and Hatice Gunes. 2023. Robotic Mental Well-being Coaches for the Workplace: An In-the-Wild Study on Form. In *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction* (Stockholm, Sweden) *(HRI '23)*. Association for Computing Machinery, New York, NY, USA, 301–310. https://doi.org/10.1145/3568162.3577003

[6] Micol Spitale, Minja Axelsson, Neval Kara, and Hatice Gunes. 2023. Longitudinal Evolution of Coachees' Behavioural Responses to Interaction Ruptures in Robotic Positive Psychology Coaching. In *2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. 315–322. https://doi.org/10.1109/RO-MAN57019.2023.10309386

[7] Aaron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu, et al. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499* 12 (2016).

Received 22 July 2024; accepted 1 August 2024