Multimodal Modeling of Coordination and Coregulation Patterns in Speech Rate during Triadic Collaborative Problem Solving

Angela E.B. Stewart
Institute of Cognitive Science
University of Colorado Boulder
Boulder, CO, USA
angela.stewart@colorado.edu

Zachary A. Keirn
Institute of Cognitive Science
University of Colorado Boulder
Boulder, CO, USA
zachary.keirn@colorado.edu

Sidney K. D'Mello
Institute of Cognitive Science
University of Colorado Boulder
Boulder, CO, USA
sidney.dmello@colorado.edu

ABSTRACT

We model coordination and coregulation patterns in 33 triads engaged in collaboratively solving a challenging computer programming task for approximately 20 minutes. Our goal is to prospectively model speech rate (words/sec) – an important signal of turn taking and active participation – of one teammate (A or B or C) from time lagged nonverbal signals (speech rate and acoustic-prosodic features) of the other two (i.e., A + B → C; A + C → B; B + C → A) and task-related context features. We trained feed-forward neural networks (FFNNs) and long short-term memory recurrent neural networks (LSTMs) using group-level nested cross-validation. LSTMs outperformed FFNNs and a chance baseline and could predict speech rate up to 6s into the future. A multimodal combination of speech rate, acoustic-prosodic, and task context features outperformed unimodal and bimodal signals. The extent to which the models could predict an individual's speech rate was positively related to that individual's scores on a subsequent posttest, suggesting a link between coordination/coregulation and collaborative learning outcomes. We discuss applications of the models for real-time systems that monitor the collaborative process and intervene to promote positive collaborative outcomes.

CCS CONCEPTS

• **Collaborative and social computing → Empirical studies in collaborative and social computing**

KEYWORDS

Collaborative problem solving; coordination; coregulation

ICMI'18, October 16-20, 2018, Boulder, CO, USA.
© 2018 Association of Computing Machinery.
ACM ISBN 978-1-4503-5692-3/18/10...$15.00.
DOI: http://dx.doi.org/10.1145/XXXXXXX.XXXXXXX

1 INTRODUCTION

Are two heads really better than one? What about three, or four, or five? Research in small group collaborative problem solving (CPS) over several decades suggests that more often than not, collaboration results in "process loss" where groups fail to achieve their full potential. This is in stark contrast to "process gain" where group interaction yields performance that exceeds the joint performance of the individual group members [36–38].

As Steiner [56] summarizes: *actual productivity = potential productivity – productivity loss due to faulty process.* Research has focused on identifying conditions where the elusive process gain can be achieved. Some of the critical variables include group size [37,38], problem structure [36,53], cohesiveness of group members in ability and motivation [15,24,34], and task constraints [8,59]. Research has sought potential causes for process losses, which can be subdivided into: (1) coordination losses, such as production blocking during collective ideation [46], the common-knowledge effect [21] (overemphasis on shared vs. individual knowledge), group-think [28] (individual members converge to the dominant view), and (2) motivation losses, such as social-loafing [29,30], evaluation apprehension [10] and free-rider effects [32].

Our present focus in on coordination processes in order to better understand and eventually prevent coordination losses. We emphasize coordination because collaboration is fundamentally about interactions among people who have thoughts, feelings, and behaviors, and who react to and influence each other's thoughts, feelings, and behaviors. Simply put, collaboration is about interactions among living and breathing people, not cold disembodied brains. It involves a host of socio-cognitive processes, such as turn taking, conversational grounding, perspective taking, emotional coregulation, behavioral mirroring, and joint action [5,6,12,48,49,66]. We hypothesize that collaborative outcomes would be productively influenced by intelligent systems that monitor these collaborative processes in real-time, triggering just-in-time interventions to improve the collaborative process. This is the long-term goal of our work. Here, we focus on real-time modeling of underlying collaborative processes, a critical step along the way.

We situate our work within a dynamical systems framework that views human interaction as a continuous and mutually adaptive process, structured by self-organization into functional synergies [7,18,22,23,44,51]. A synergy occurs when interacting components can function as a single unit. It arises as a system's componential degrees of freedom become loosely coupled and mutually constrain each other, resulting in a dramatic and temporary reduction in the shared set of possibilities, allowing for more stable and coordinated forms of behavior [50,54]. Accordingly, for effective collaboration, individuals' behavioral patterns, including those that map onto cognitive and affective states, are expected to come together as dynamic couplings of coordination and coregulation. These couplings are not simply aggregated behaviors, but are emergent patterns that reflect the activity of the system as a whole. Specifically, coordination refers to (near) concurrent bidirectional linkages of behavior (e.g., facial expressions, eye gaze) amongst interacting partners; coregulation refers to coupling at greater temporal lags and captures asymmetrical and symmetrical leader/follower patterns. Together, the processes sustain long-term temporal dependencies across the entire interaction, giving rise to patterns of global stability and complexity. These are also expressed across multiple interacting channels and index and maintain higher-order components of successful collaboration, including effective communication, negotiation/coordination, and maintaining team function [57].

We propose predictive modeling of coordination and coregulation in triads remotely engage in a collaborative problem solving task. We focus on modeling speech rate which indexes active participation and turn taking [31], which are critical for successful collaboration [12,31]. More specifically, coordinated speech rate, as well as speech and intensity, have been associated with features of agreeableness and well-functioning conversations [41,44]. The structure of how people regulate turn-taking also provides insights into communication and active participation via the maintenance of conversational flow, efficient informational exchange, experiential quality [22,51], and indexing engagement in the timing of turns to anticipate turn completions [7].

1.1 Related Work

Researchers have traditionally investigated coordination in terms of *synchrony*, where two or more people in a shared situation, task, or conversation coordinate behaviors in order to maintain common ground, establish social bonding, and improve quality of social interactions [9,23]. Some methods to analyze synchrony include time lagged correlations [9,14], recurrence analysis [9,14], and coupled oscillatory models [35] (see [9] for a review). More recent methods, such as complexity matching [1], go beyond occurrences of the observed behavior and examine power law distributions of behaviors. For example, Abney et. al. [1] found evidence for complexity matching in acoustic onsets during affiliative dyadic, but not in argumentative conversations.

Taking a multimodal approach, Duran and Fusaroli [14] examined coordination of head movements and speech rate during deceptive conversations. They hypothesized that deceivers maintain believability through heightened attention with their conversational partner, achieved by maintaining synchrony. They used window lagged cross correlations to analyze synchrony in head movements at short time scales (less than 1000 milliseconds) and cross recurrence quantification analysis to understand global patterns in synchrony of speech rate. They found that deceivers synchronized with their conversational partners' head movements at short lags (between 0 and 1000 milliseconds), concluding that deceivers closely follow the lead of their partner and anticipate cues. Additionally, when deceiving a conversational partner, speech rate was adapted to the partner as the conversation changed.

In contrast to the aforementioned analytic approaches, which compute indices of synchrony from dyadic data, predictive model-based approaches emphasize the use of behaviors of one or more conversational participants to predict the behaviors of a different target partner. We could only find two such studies. First, Feng et. al. [18] used variational autoencoders and deep neural networks, trained on data collected during Skype conversations, to generate facial expressions of an avatar from a human interlocutor's facial cues. Second, Grafsgaard et. al. [23] used long-short term memory networks to model facial expressions and motion features of heterosexual romantic couples using the behaviors of the male to predict the behaviors of the female and vice versa. They found that their model-based measure of synchrony revealed unique insights compared to a naïve analytic approach of simply correlating the partners' raw time series. We adopt a similar approach here.

1.2 Contribution and Novelty

The key idea of our work is that because coordination and coregulation are fundamentally about dependencies across participants, predicting behaviors of one partner from the behaviors of the others is a more direct test of such dependencies than simply quantifying them as in the analytic approach. Accordingly, in this work, we adopt a predictive approach to modeling speech rate of one conversational partner from behaviors of his/her teammates. Specifically, we use data from two teammates to predict speech rate of the third team member (i.e., A + B → C; A + C → B; B + C → A). Importantly, the models are trained using data from different teams to generate predictions for the target team. We hypothesize that the fit between the model predictions and the original time series (A_{pred} vs. A_{orig}; B_{pred} vs. B_{orig}; C_{pred} vs. C_{orig}) reflect *global* patterns of coordination and coregulation (because the models are trained on different teams). In contrast, the analytic approach would simply compute measures of synchrony from the original time series, thereby reflecting more *local* patterns.

To our knowledge, this is the first attempt to predict speech rate of an individual solely from multimodal behavioral inputs of his/her conversational partners. There is related work on multimodal end-of-turn and next speaker prediction [11,33], but this research does not use behavioral inputs from the other people in the conversation, which is a critical component of our approach.

Additionally, compared to other related work [18,23], our approach is multimodal, focuses on speech rather than facial expressions, and considers coregulation by lagging input time series to make future predictions compared to mere coordination (synchrony at lag 0). We also consider triadic interactions because they provide a rich interaction context as synchrony occurs between dyads within the triad, or all three team members.

2 DATA COLLECTION

2.1 Participants

Participants were 111 (63.1% female, average age = 19.4 years) undergraduate students from a medium-sized private Midwestern university, who were compensated with course credit. Participants were 74.8% Caucasian, 9.9% Hispanic/Latino, 8.1% Asian, 0.9% Black, 0.9% American Indian/Native Alaskan, 2.7% other, and 2.7% did not report ethnicity. Participants were assigned to 37 teams of three based on scheduling constraints. Nineteen participants from ten teams (27%) indicated they knew at least one person from their team prior to participation. The only inclusion criterion was no previous experience with computer programming; none of the participants were excluded on this basis.

Four teams were removed because at least one participant in the team was missing an audio file. One team was missing a screen recording due to equipment failure, but was still used (see Section 3.1). Thus, we analyzed 33 teams.

2.2 Study Protocol

Participants were randomly assigned to one of three computer-equipped rooms in a lab. Each computer had a webcam with a microphone so participants could see and hear each other, facilitated though Zoom's video-conferencing and screen sharing capabilities(https://zoom.us). Participant audio was recorded on separate streams. The screen content was also recorded using Zoom's built-in features (see Figure 1).

The task involved completing an introductory and a collaborative problem solving activity using code.org's (an online resource that teaches basic computer programming principles) Minecraft-themed Hour of Code [60] . Hour of Code uses Blockly [20], a visual programming language that represents lines of code (such as loops) as syntactically-correct interlocking blocks. One participant (designated participant A) was randomly assigned to interact with the environment and shared their screen content with the other participants (designated B and C). This was done due to technical constraints with the code.org web-based interface which was not inherently designed to support collaborations.

In an introductory task, teams completed five lessons and watched three accompanying videos that taught basic programming principles along with instructions on how to use the coding environment. Participants were instructed to collaborate as a team to complete this task within 20 minutes.

After completing the introductory task, the screen share was disabled, and participants individually rated their level of satisfaction with their team's: (1) "*performance* at completing the lessons;" (2) how well their team "*communicated* with each other;" (3) how well their team "*cooperated* to complete the lessons;" and (4) how "*agreeable* my teammates are;" Participants used a very dissatisfied (1) to very satisfied (6) scale for these ratings.

The main CPS activity involved a challenging programming task where teams had 20 minutes to build a 4×4 brick building with the following constraints: use at least one if statement; use at least one repeat loop; build at least three bricks over water; and use 15 blocks of code or less. The same team member who controlled the interaction with the environment during the introductory phase also controlled the interaction during the coding challenge.

After completing the challenging programming task, participants individually completed the same subjective assessments of their team's performance, communication, cooperation, and agreeableness. Finally, participants individually completed a ten-item researcher-created multiple-choice test to assess their conceptual knowledge of coding concepts (such as repeat loops and if statements).

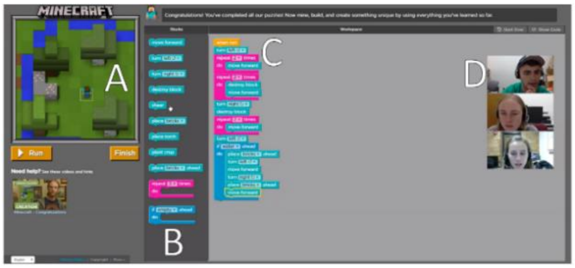

MINECRAFT
Congratulations! You've completed all our puzzles! Now mine, build, and create something unique by using everything you've learned so far.
A
B
C
D
Run
Finish
Need help?

Figure 1: Minecraft-themed Hour of Code. Participants could (A) visualize the results of running their code, (B) choose code blocks from a code bank, (C) generate solutions to the task, and (D) see their team's faces.

3 MODEL DESIGN AND TRAINING

We model speech rate of each participant using behavioral features (speech rate, acoustic-prosodic features) of the other team members, as well as team-level task context features. We did not include facial features due to considerable missing data when the face of one of the teammates could not be tracked in the video stream. We also focus on the challenging programming task because the main purpose of the introductory activity was to familiarize participants with the environment and their teammates.

3.1 Feature Processing

We used the IBM Watson Speech to Text service [61] to generate transcriptions of individual audio recordings, from which we computed speech rate (words per second) for each second of the collaboration. If a word spanned multiple seconds, we assigned it to the second in which it started.

We used the openSMILE toolkit [16] to extract the following acoustic-prosodic features over 10 millisecond windows: fundamental frequency, loudness, center frequency of the first through third formants, first through third formant amplitudes, harmonics to noise ratio, jitter, and shimmer.

We used the screen recording to extract high-level task context features as a measure of the teams' actions within the environment (log files were not available). We focused on two areas of interest (AOI) – the code runtime environment (A in Figure 1) and the code bank and workspace (B and C in Figure 1) – and used a validated motion estimation algorithm [58] to compute the amount of change in each area. Change in the code bank and workspace AOI indicated how many edits the team made to their solution, whereas the code runtime AOI indicated attempts to test their code.

We computed one binary validity feature for whether speech rate and acoustic-prosodic features could be calculated and another for whether task context features could be computed. These features were only invalid when the relevant data file was missing or incomplete (see Section 2.1). In all, there were 16 features per participant: one speech rate feature, 11 acoustic-prosodic features, two task context features, and two validity features.

3.2 Data Aggregation

Our target outcome (words per second) was computed at a 1s granularity, which we deemed appropriate for these interactions. Because acoustic-prosodic and task context features were computed at different rates (100 Hz and 25 frames per second respectively), we averaged each of these features across non-overlapping 1s windows. We also considered a coarser 3s granularity by aggregating the 1s time series across 3s non-overlapping windows for speech rate, acoustic-prosodic, and task context features (see Figure 2). For the binary validity features, we computed the sum over the 3s window and transformed sums greater than zero to a binary validity of one. The 1s and 3s aggregations yielded time series of approximately 1200 and 400 feature vectors (per participant), respectively, across the 20-minute CPS phase. Time series length varied slightly for teams who completed the task before the allotted 20 minutes.

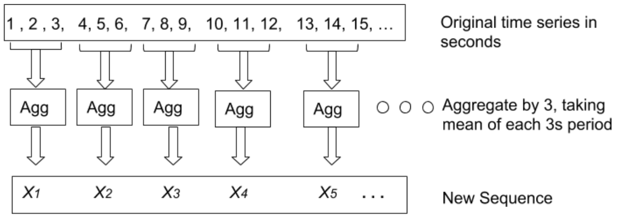


Figure 2: Aggregation from a 1s to 3s time series.

3.3 Neural Network Modeling

We built neural network models to predict the speech rate of one teammate from behavioral signals (speech rate and acoustic-prosodic) of the other two (i.e., A + B → C) along with team-level task context features. We compared two neural network model types using Keras with TensorFlow [62]. The first was a feed-forward neural network (FFNN) [63] with a single fully connected dense layer. The second was a long-short term memory network (LSTM) which is a special type of recurrent neural network that can learn long term dependencies [26] by selectively retaining and forgetting information across input sequences. Both network types had a single hidden layer. We chose to use FFNNs and LSTMs because they have been applied to similar data and modalities [17,45,47]. We expected LSTMs to outperform FFNNs on data with time dependencies such as turn taking in speech.

The LSTM was trained on sequences of inputs from the 1s and 3s aggregations. The general form for an input sequence of length m would be $\mathrm{X}^{\mathrm{a}}_{\mathrm{k-m+1}}, \mathrm{X}^{\mathrm{a}}_{\mathrm{k-m+2}}, \dots, \mathrm{X}^{\mathrm{a}}_{\mathrm{k}}$ to predict Y^{a}_{k+L} at lag L where a represents different aggregation windows (a=1s or a=3s) and k is the sequence index. For example, an m of 2, L of 1, and a of 1s indicates predicting the next time point from the previous two time points in a 1s aggregated time series. Figure 3 shows example input sequences of length m=3 and predicted outputs at various lags. The FFNN takes only a single value X^{a}_{k} as input to predict the output at Y^{a}_{k+L}. For example, using input X^{a}_{2} to predict output Y^{a}_{3} at lag L=1.

Note that we refer to lags instead of leads because we envision the input time series *lagging* behind the output time series. Further, for a given lag, the extent to which we are predicting into the future pertains to the aggregation window length a. For a fixed lag of 2, an a of 1s would indicate predicting 2s in the future, but an a of 3s would involve predicting ahead by 6s.

Input sequence length $m = 3$			Predicted output at L				
			L = 0	L= 1	L = 2	L = 3	…
X_2	X_3	X_4	Y_4	Y_5	Y_6	Y_7	…
X_3	X_4	X_5	Y_5	Y_6	Y_7	Y_8	…
X_4	X_5	X_6	Y_6	Y_7	Y_8	Y_9	…
…	…	…	…	…	…	…	…

Figure 3: Example input and output sequences

We used team-level 10-fold cross validation (using scikit-learn [64]) to train and test our models. Within each fold, we further split the data into 60% training, 30% validation, and 10% testing.

All features were z-scored and then normalized to a [-3, 3] range per fold. We used the training data to compute the statistics needed for the z-scoring and normalization (mean, standard deviation, max, min), which were subsequently applied to the validation and testing sets. Preliminary results indicated that z-scoring with [-3, 3] normalization slightly outperformed [-1, 1] normalization, greatly outperformed [0, 1] normalization, and was equivalent to z-scoring without normalization. We selected z-scoring with [-3, 3] normalization to address missing values, which were replaced with a value of 5, chosen to be outside the normalized range. This only occurred for the task context features of one team that was missing a screen recording (Section 2.1). This binary input mask to indicate if data was missing for each modality (Section 3.1) was shown to be useful for training LSTMs with missing data [42].

Both networks utilized a single hidden layer with 32 units and leaky rectified linear unit activation function, which has been shown to improve performance and reduce training time in deep learning applications [43]. We chose 32 units after comparing validation loss across 8, 16, 32, and 64 units revealed that 32 units was adequate. Similarly, we compared networks with 1, 2, and 3 hidden layers and selected models with a single hidden layer as they achieved equitable performance compared to deeper networks. Thus, the final models had a single layer of 32 hidden units. Further, the LSTM models utilized a sequence length of 3s after experimentation (see below).

Neural networks use gradient descent and back propagation to update the weights during each complete pass of the training (referred to as a training epoch). At each epoch, a loss function (mean squared error) was computed and the weights were updated via backpropagation. We used an adaptive learning rate algorithm, NADAM [13], to tune the learning rate. We fixed the number of training epochs to 50 since the models converged within 50 epochs.

We experimented with batch normalization [27] , *l2* weight regularization [65], and dropout [55] to prevent overfitting. We found that dropout had no discernible impact when combined with the other two methods. Additionally, we found that the default parameters used in the Keras implementations of batchnorm and kernel regulation were adequate for our data.

4 RESULTS

Our key outcome measure is prediction accuracy, computed as the correlation coefficient between the observed and predicted speech rate time series. We used the nonparametric Spearman rank-order correlation as opposed to the parametric Pearson product moment correlation because the time series are zero-inflated (i.e. when the target participant does not speak), thereby violating normality assumptions. To ensure fair comparisons across temporal granularity (i.e., 1s and 3s level of aggregation), we averaged correlations across lags for the 1s aggregation models to align with the lags of the 3s aggregation models. For example, we averaged the first, second, and third seconds of the 1s aggregated time series and compared it to the first data point in the 3s aggregated time series. Similarly, we averaged the fourth, fifth, and sixth seconds of the 1s aggregated time series and compared it to the second data point in the 3s aggregated time series. No averaging was done for lag 0.

Figure 4 provides a histogram of Spearman correlations for all observations, suggesting a positive skew but no notable outliers.

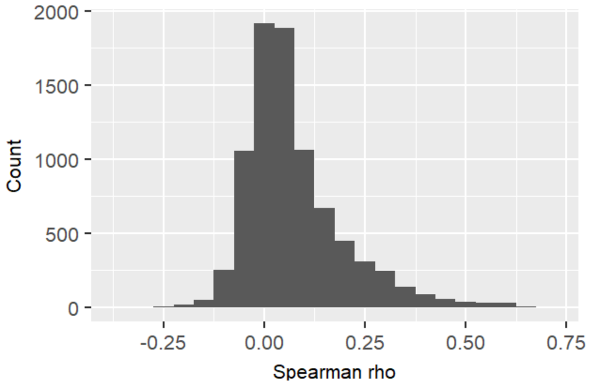


Figure 4: Histogram of correlations across all observations

Due to the repeated (multiple observations per team) and nested structure of the data (participants nested within teams), we used linear mixed effects regressions to model the data [4]. We included the number of voiced segments as a fixed effects covariate in all models because it was weakly correlated with prediction accuracy (Pearson *rs* = .133 and .176 for 1s and 3s aggregation, respectively). Team was included as a random intercept in all models.

We used the `lme4` package in R for the linear mixed effects models [4], the `car` package (Companion to Applied Regression) for significance testing of main effects and interactions [19], and the `emmeans` (estimated marginal means) package [39] for pairwise comparisons and to probe interactions [40].

Selecting a model: Network type and aggregation level. Our first step was to choose a network type and level of aggregation. Accordingly, we regressed prediction accuracy (Spearman's rho) on the three-way interaction between network type (FFNN vs. LSTM) × aggregation level (1s vs. 3s) × lag (0s, 3s, 6s, 9s). Modality (speech rate, speech rate + task context, speech rate + task context + acoustic-prosodic) and team member (A, B, or C) were included as categorical fixed effects; these are examined in more detail once a model is selected. The three way interaction was not statistically significant (p = .521), suggesting similar results across lags. However, there was a significant interaction between aggregation level and network type ($\chi^2(1)$ = 4.39, p = .036, Figure 5). Pairwise comparisons, averaging across lag, modality, and team member, indicated that there was no significant difference (p = .619) between network types for the 3s aggregation, but LSTMs outperformed FFNNs (p < .001) for the 1s aggregation. Overall, prediction accuracy was also higher (p < .001) for the 3s compared to the 1s aggregation level. We focused on 3s aggregation and LSTMs for all subsequent analyses as they have a slight advantage over FFNNs. We also expanded the model to 12s and 15s lags to investigate how far out into the future we could predict speech rate.

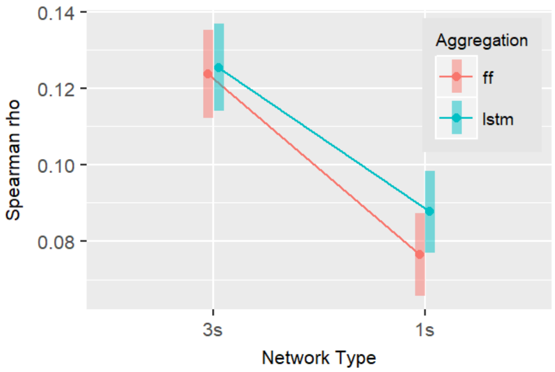
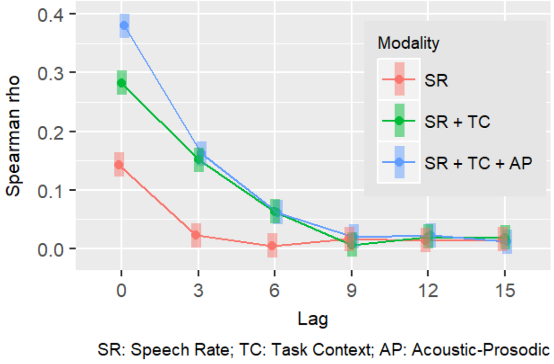


Figure 5: Interaction between network and aggregation level

Effects of modality and team member. We regressed prediction accuracy on the two-way interaction between modality × lag. We did not consider the three-way modality × lag × team member interaction as this is not of theoretical interest. Instead, team member was included as a fixed main effect, which was statistically significant ($\chi^2(2)$ = 49.7, p < .001). Pairwise comparisons with a Tukey correction for multiple comparisons indicated that prediction accuracies for the two team members who did not control the interface (i.e., B and C) were significantly higher (p < .001) than the team member (i.e., A) who controlled the interface. Further, fit for team member C was also higher (p < .001) than team member B (i.e., C > B > A). The difference in prediction accuracy between participants B and C compared to participant A might be attributable to more speech production by participant A ($\chi^2(2)$ = 38.0, p < .001) compared to B and C (p's < ,001). However, it would not explain the C > B difference because these participants produced equivalent speech (p = .938).

The modality × lag interaction was also significant, ($\chi^2(10)$ = 321, p < .001, see Figure 6). Pairwise comparisons (with a Tukey adjustment) across the three modalities for each lag indicated that adding information on the task context to speech rate increased prediction accuracy for lags 0s, 3s, and 6s (p's < .001), beyond which there were no statistical differences (p's > .714). Similarly, adding acoustic-prosodic features to speech rate and task context only improved fit for lag 0s (p < .01); there were no detectable differences for the other lags (p's > .503). Modality had no affect beyond lag 9s, upon which the correlations were basically zero.



SR: Speech Rate; TC: Task Context; AP: Acoustic-Prosodic

Figure 6: Interaction between modality and lag (in seconds)

Because the overall best results were obtained for the LSTM model with speech rate, task context, and acoustic-prosodic features, subsequent analyses focus on this model. Recall that this model aggregates data in 3s intervals and the LSTMs were trained with a sequence length of 3 resulting in 9s of history. We also experimented with sequence lengths of five (i.e. 15s of history) and seven (i.e. 21s of history), but the results were virtually identical (p = .619 for main effect of sequence length; p = .924 for sequence length × lag interaction). Thus, we proceeded with a sequence length of 3 (9s of history).

Comparison with shuffled baseline. We trained surrogate models by shuffling the output time series per team while preserving the temporal integrity of the input time series. For example, we would shuffle participant C's time series in the A + B → C model to produce $C_{shuffle}$. We trained an LSTM on A + B→ $C_{shuffle}$ and compare its prediction accuracy to the prediction accuracy obtained with the original non-shuffled time series. This form of shuffling provides an important baseline because it preserves the central tendency of the speech rate distribution (i.e., mean and variance) while breaking temporal dependencies.

We regressed prediction accuracy (Spearman rho) on a two-way interaction between shuffle (yes vs. no) × lag (with team member as a fixed main effect). There was a statistically significant interaction, $\chi^2(5)$ = 396, p < .001. Pairwise comparisons indicated that the non-shuffled model significantly (p < .001) outperformed the shuffled model for lags 0s, 3s, and 6s but not for lags 9s, 12s, and 15s, see Figure 7).

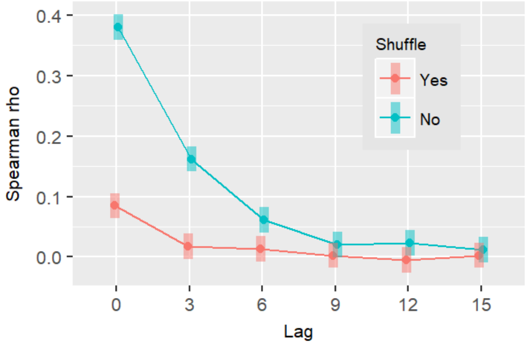


Figure 7: Interaction between shuffle and lag (in seconds).

We also note that fit for the shuffled models was approximately zero past lag 0s, whereas the non-shuffled models maintained a nonzero correlation for lags 0s (rho = .380), 3s (rho = .162), and 6s (rho = .062), beyond which they were basically zero (i.e., .021, .023, and .012 for lags 9s, 12s, and 15s, respectively). Thus, the models could predict speech rate up to 6s in the future.

Comparison with self-models. The current models utilize data from a target participant's teammates along with information from the task context to predict his/her own speech rate. How do these *peer* models (e.g., A + B + task context → C) compare to *self*-models (e.g., C + task context → C)? We addressed this question by training LSTM models to predict the target's future speech rate from *lagged* versions of his or her own speech rate, acoustic-prosodic features, and task context features (e.g., C_{lagged}→ C_{future}). Prediction accuracy of these self-models can be considered to be an upper bound on what can be achieved with our approach.

We used the same statistical model to analyze the data, with a focus on the input source (self vs. peer) × lag interaction term. We did not include lag 0s in the analysis as it essentially equates to training and testing on the same data for the self-models. The results (see Figure 8) indicate a significant interaction, $\chi^2(4)$ = 84, p < .001. As expected, the self-model outperformed (p < .001) the peer model, and its main advantage emerged for the earlier lags, rapidly decreasing beyond lag 12s.

Prediction accuracies for the self and peer models were moderately correlated (r's between .355 and .520) for lags 0s to 9s and more weakly correlated (r's .197 and .198) for lags 12s and 15s. Thus, the models appear to be tapping related, but not redundant information.

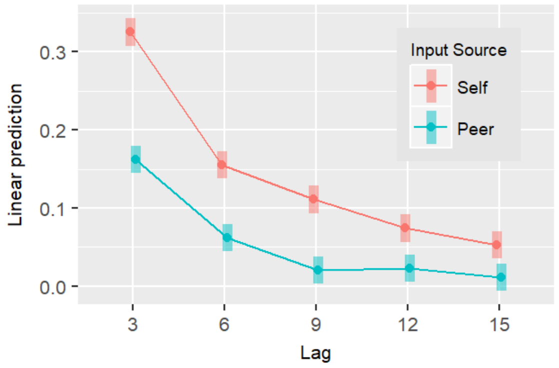


Figure 8: Interaction between input source and lag (in secs)

Predicting individual outcomes. We examined whether the accuracy of our speech rate predictions were related to collaborative problem solving outcomes. Because the models generate participant-level predictions of speech rate of each team member, we focused on each participant's objective posttest score, and their subjective assessments of their team's performance, communication, cooperation, and agreeableness. We averaged each participant's communication, cooperation, and agreeableness scores since they were strongly correlated (Cronbach's alpha = .89). We separately regressed each of these outcome variables on prediction accuracy with team member (A, B, and C) and number of voiced segments as fixed effects covariates. Team was an intercept-only random effect. We focused on the lag 0 model as these yielded the best prediction accuracy, and thus is most reliable.

The results indicated no significant effect of prediction accuracy on subjective perceptions of performance (p = .617) or the average of communication, cooperation, and agreeableness (p = .460). However, prediction accuracy significantly predicted posttest scores (B = 1.40, SE = .704, $\chi^2(1)$ = 3.93, p = .047). Participants whose speech rate could be more accurately predicted from their peers' data had higher posttest scores (Figure 9). The same effect was not observed (p = .181) for the self-model.

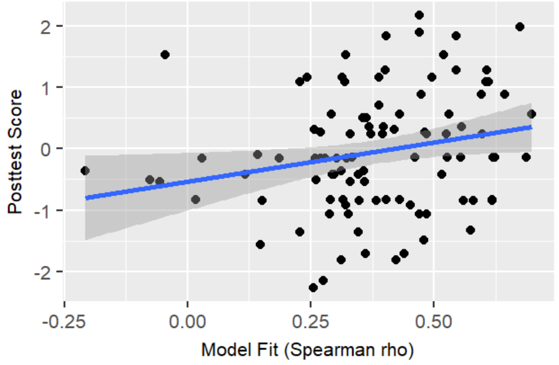
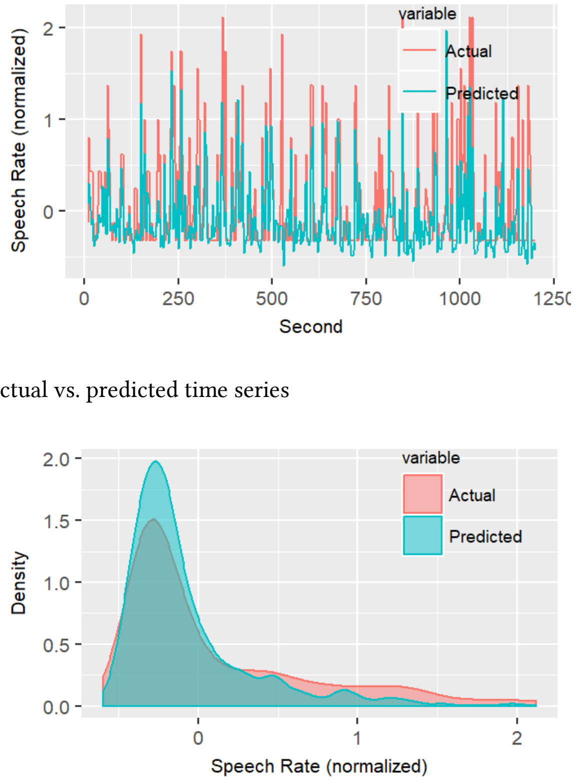


Figure 9: Correlations between prediction accuracy and posttest score

Example output. Figure 10 provides an example of the predicted and actual time series for one participant for which the model was particularly accurate (Spearman rho = .700).



(A) Actual vs. predicted time series

(B) Density plots of actual vs. predicted distributions

Figure 10: Sample results for one team member

5 DISCUSSION

We adopted a predictive approach to modeling coordination and coregulation during collaborative problem solving, hypothesizing that the predictive models capture global patterns of collaboration compared to existing analytic approaches. Accordingly, we used speech rate and acoustic-prosodic features of two teammates and information on task context to model the speech rate of the third teammate, training the models on data from unrelated teams.

5.1 Main Findings

We found that a combination of speech rate, acoustic-prosodic features, and task context features outperformed unimodal or bimodal models. The biggest boost was obtained by the addition of task context features, which encode when a team makes changes to their solution and runs it. We hypothesize that these features capture knowledge-building discourse [25], which occur when teams collectively generate, communicate, and iteratively refine ideas [25], a pattern somewhat enforced by a combination of concrete goals of the coding task and instant feedback (i.e. teams run their code). The addition of acoustic-prosodic features increased prediction accuracy at lag 0s, presumably because they communicate nonverbal cues that the conversational floor is open.

We could successfully predict speech rate up to 6s into the future, with the best result occurring at lag 0s, presumably due to the model encoding turn-taking dynamics. In conversation, people usually do not speak at the same time [14], making information about speaking patterns closer to the target a more accurate description of the unfolding conversation. That said, a lack of speech of two partners does not imply that the third will speak because there are periods of silence, especially when a solution is being tested or during periods of ideation.

As expected, models with inputs of a person's own speech rate (self-models) outperformed those with only inputs of that person's teammates (peer models). That said, the results for the self-models were far from perfect with correlations slightly exceeding 0.3 at lag 3s. This provides an important upper bound of what might be achievable with these models, data, and features due to the multiple influences on whether a person will speak and at what rate. This is reflected by the fact that the prediction accuracy was lower for the participant who controlled the interface, ostensibly because there are additional degrees of freedom for these participants. Further, recall that the goal is not to merely predict speech rate but to model team collaboration dynamics. The fact that we could achieve correlations as high as 0.380 suggest that the model is capturing interesting aspects of the dynamics of the unfolding collaboration.

The accuracy of the model's predictions of speech rate predicted how much an individual would learn as a result of the collaboration. This result is possibly related to the establishment of common ground and success in coordinating action with the group, which are related to successful collaborative outcomes [3,31]. Further, this result provides external evidence that the models are picking up collaborative patterns among team members because more cohesive teams should theoretically be more predictable.

It is particularly notable that our models were able to predict speech rate at all because virtual collaborations lack the richness of social cues present in face-to-face interaction [2,52], which can lead to impaired ability to coordinate action, read social cues, and may also reduce engagement [52]. It is possible that teams that overcame the challenges associated with virtual collaboration established more coupled and regulatory behaviors, which led to more accurate modeling and better learning outcomes.

There were also differences among team roles, as we were more accurate at predicting speech rate of team members who were not controlling the interaction (i.e., B and C) compared to those who were (i.e., A). This suggests inherent differences in how coordination and coregulation is manifested, depending on the role a person takes in a collaboration. What is surprising, however, is that the models were also more accurate at predicting the speech rate of participant C compared to B despite both having the same role. This is a puzzling finding worthy of replication.

5.2 Applications

Our method of modeling speech rate of an individual in a virtual collaborative context is fully automated, generalizable across teams (due to our cross-validation method), and is, to some extent, applicable to new tasks (due to our method of generating task context features without log files). Thus, it can be used in a real-time system that supports productive virtual collaboration for triads. In particular, our multimodal models were able to prospectively predict speech rate, up to six seconds into the future, allowing for regulation of behavior before it occurs. For example, team members who might interrupt other team members could be prompted not to do so prior to exhibiting that behavior. The model predictions of future verbal contributions could be used to determine when to encourage participation from team members that might not speak as often as others.

Finally, prediction accuracy was positively related to learning outcomes, leading us to conclude that the models are to some extent indexing positive team dynamics. Therefore, a real-time system could use prediction accuracy (comparing predicted speech rate to actual speech rate over the past few seconds) to determine when and how to intervene. Teams with a low prediction accuracy might be struggling to effectively coordinate and coregulate and could receive an intervention to help them collaborate effectively. Teams receiving high prediction accuracy should be left alone but should be monitored for notable decreases, which would be used to trigger appropriate interventions.

5.3 Limitations and Future Work

Our work has limitations that should be addressed in the future. For one, the dataset is small and was collected at a single university with little ethnic, age, or cultural diversity, and focused on a single task. We were also unable to model facial features due to large amounts of missing data, thereby missing an important social cue. Additionally, this work should be expanded to include the content of what was said, in addition to speech rate. We are currently investigating these limitations by collecting a second dataset with quality facial expression tracking, from multiple universities, and across multiple collaborative tasks. We are also collecting and analyzing additional modalities, such as eye gaze, linguistic information, and peripheral physiology.

5.4 Conclusion

We modeled coordination and coregulation patterns during collaborative problem solving using a predictive approach where the behavior of one individual was predicted from the behavior of the other individuals in a triad. We were able to predict when a person will speak up to six seconds in the future and model accuracy was predictive of collaborative outcomes. The next step is to leverage the models to trigger interventions that dynamically support collaborative processes in virtual teams.

6 ACKNOWLEDGEMENTS

This research was supported by the National Science Foundation (NSF DUE 1745442) and the Institute of Educational Sciences (IES R305A170432). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the funding agencies.

7 REFERENCES

[1] Drew H. Abney, Alexandra Paxton, Rick Dale, and Christopher T. Kello. 2014. Complexity matching in dyadic conversation. *J. Exp. Psychol. Gen.* 143, 6 (2014), 2304–2315.

[2] Richard Alterman and Kendall Harsch. 2017. A more reflective form of joint problem solving. *Int. J. Comput. Collab. Learn.* 12, 1 (March 2017), 9–33. DOI:https://doi.org/10.1007/s11412-017-9250-1

[3] Brigid Barron. 2000. Achieving coordination in collaborative problem-solving groups. *J. Learn. Sci.* 9, 4 (2000), 403–436. DOI:https://doi.org/10.1207/S15327809JLS0904_2

[4] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2015. Fitting Linear Mixed-Effects Models Using lme4. *J. Stat. Softw.* 67, 1 (2015), 1–48. DOI:https://doi.org/10.18637/jss.v067.i01

[5] Emily A. Butler and Ashley K. Randall. 2013. Emotional Coregulation in Close Relationships. *Emot. Rev.* 5, 2 (April 2013), 202–210. DOI:https://doi.org/10.1177/1754073912451630

[6] Herbert H. Clark and Susan E. Brennan. 1991. Grounding in communication. *Perspect. Soc. Shar. Cogn.* 13, 1991 (1991), 127–149.

[7] Jennifer Coates. 1994. No gap, lots of overlap; turn-taking patterns in the talk of women friends. In *Researching Language and Literacy in Social Context: A Reader*, David Graddol, Janet Maybin and Barry Stierer (eds.). Multilingual Matters, 177–191.

[8] Traci Y. Craig and Janice R. Kelly. 1999. Group cohesiveness and creative performance. *Gr. Dyn. Theory, Res. Pract.* 3, 4 (1999), 243–256. DOI:https://doi.org/10.1037/1089-2699.3.4.243

[9] Emilie Delaherche, Mohamed Chetouani, Ammar Mahdhaoui, Catherine Saint-Georges, Sylvie Viaux, and David Cohen. 2012. Interpersonal synchrony: a survey of evaluation methods across disciplines. *IEEE Trans. Affect. Comput.* 3, 3 (October 2012), 349–365. DOI:https://doi.org/10.1109/T-AFFC.2012.12

[10] Michael Diehl and Wolfgang Stroebe. 1987. Productivity loss in brainstorming groups: Toward the solution of a riddle. *J. Pers. Soc. Psychol.* 53, 3 (1987), 497–509. DOI:https://doi.org/10.1037/0022-3514.53.3.497

[11] Alfred Dielmann, Giulia Garau, and Hervé Bourlard. 2010. Floor holder detection and end of speaker turn prediction in meetings. In *Proceedings of the International Conference on Speech and Language Processing, Interspeech.*

[12] Pierre Dillenbourg. 1999. What do you mean by collaborative learning? *Collab. Cogn. Comput. Approaches* 1, 6 (1999), 1–19. DOI:https://doi.org/10.1.1.167.4896

[13] Timothy Dozat. 2016. Incorporating nesterov momentum into adam. In *Proceedings of the International Conference on Learning Representations.*

[14] Nicholas D. Duran and Riccardo Fusaroli. 2017. Conversing with a devil's advocate: Interpersonal coordination in deception and disagreement. *PLoS One* 12, 6 (June 2017), e0178140. DOI:https://doi.org/10.1371/journal.pone.0178140

[15] Charles R. Evans and Kenneth L. Dion. 1991. Group cohesion and performance: A meta-analysis. *Small Gr. Res.* 22, 2 (1991), 175–186. DOI:https://doi.org/10.1177/1046496412468074

[16] Florian Eyben, Felix Weninger, Florian Gross, and Björn Schuller. 2013. Recent developments in openSMILE, the Munich open-source multimedia feature extractor. In *Proceedings of the 21st ACM international conference on Multimedia (MM '13)*, 835–838. DOI:https://doi.org/10.1145/2502081.2502224

[17] Bo Fan, Lijuan Wang, Frank K. Soong, and Lei Xie. 2015. Photo-real talking head with deep bidirectional LSTM. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4884–4888. DOI:https://doi.org/10.1109/ICASSP.2015.7178899

[18] Will Feng, Anitha Kannan, Georgia Gkioxari, and C. Lawrence Zitnick. 2017. Learn2Smile: Learning non-verbal interaction through observation. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4131–4138. DOI:https://doi.org/10.1109/IROS.2017.8206272

[19] John Fox, Michael Friendly, and Sanford Weisberg. 2013. Hypothesis tests for multivariate linear models using the car package. *R J.* 5, 1 (2013), 39–52.

[20] Neil Fraser. 2015. Ten things we've learned from Blockly. In *Proceedings of the 2015 IEEE Blocks and Beyond Workshop*, 49–50. DOI:https://doi.org/10.1109/BLOCKS.2015.7369000

[21] Daniel Gigone and Reid Hastie. 1993. The common knowledge effect: Information sharing and group judgment. *J. Pers. Soc. Psychol.* 65, 5 (1993), 959–974.

[22] Jamie C. Gorman, Nancy J. Cooke, Polemnia G. Amazeen, and Shannon Fouse. 2012. Measuring Patterns in Team Interaction Sequences Using a Discrete Recurrence Approach. *Hum. Factors J. Hum. Factors Ergon. Soc.* 54, 4 (August 2012), 503–517. DOI:https://doi.org/10.1177/0018720811426140

[23] Joseph Grafsgaard, Nicholas Duran, Ashley Randall, Chun Tao, and Sidney D'Mello. 2018. Generative multimodal models of nonverbal synchrony in close relationships. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, 195–202. DOI:https://doi.org/10.1109/FG.2018.00037

[24] Stanley M. Gully, Dennis J. Devine, and David J. Whitney. 1995. A meta-analysis of cohesion and performance. Effects of level of analysis and task interdependence. *Small Gr. Res.* 26, 4 (November 1995), 497–520. DOI:https://doi.org/10.1177/1046496495264003

[25] Cindy E. Hmelo-Silver and Howard S. Barrows. 2008. Facilitating collaborative knowledge building. *Cogn. Instr.* 26, 1 (January 2008), 48–94. DOI:https://doi.org/10.1080/07370000701798495

[26] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (November 1997), 1735–1780. DOI:https://doi.org/10.1162/neco.1997.9.8.1735

[27] Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning*, 448–456.

[28] Irving Lester Janis. 1982. *Groupthink: Psychological studies of policy decisions and fiascoes.* Houghton Mifflin, Boston.

[29] Steven J. Karau and Kipling D. Williams. 1993. Social loafing: A meta-analytic review and theoretical integration. *J. Pers. Soc. Psychol.* 65, 4 (1993), 681–706.

[30] Norbert L. Kerr. 1983. Motivation losses in small groups: A social dilemma analysis. *J. Pers. Soc. Psychol.* 45, 4 (1983), 819–828. DOI:https://doi.org/10.1037/0022-3514.45.4.819

[31] Norbert L. Kerr and R. Scott Tindale. 2004. Group performance and decision making. *Annu. Rev. Psychol.* 55, 1 (February 2004), 623–655. DOI:https://doi.org/10.1146/annurev.psych.55.090902.142009

[32] Norbert L Kerr and Steven E Bruun. 1983. Dispensability of member effort and group motivation losses: Free-rider effects. *J. Pers. Soc. Psychol.* 44, 1 (1983), 78–94.

[33] Iwan de Kok and Dirk Heylen. 2009. Multimodal end-of-turn prediction in multi-party meetings. In *Proceedings of the 2009 International Conference on Multimodal Interfaces* (ICMI-MLMI '09), 91–98. DOI:https://doi.org/10.1145/1647314.1647332

[34] Claus W. Langfred. 1998. Is group cohesiveness a double-edged sword? *Small Gr. Res.* 29, 1 (February 1998), 124–143. DOI:https://doi.org/10.1177/1046496498291005

[35] Edward W. Large and Mari Riess Jones. 1999. The dynamics of attending: How people track time-varying events. *Psychol. Rev.* 106, 1 (1999), 119–159. DOI:https://doi.org/10.1037/0033-295X.106.1.119

[36] Patrick R. Laughlin and Alan L. Ellis. 1986. Demonstrability and social combination processes on mathematical intellective tasks. *J. Exp. Soc. Psychol.* 22, 3 (May 1986), 177–189. DOI:https://doi.org/10.1016/0022-1031(86)90022-3

[37] Patrick R. Laughlin, Erin C. Hatch, Jonathan S. Silver, and Lee Boh. 2006. Groups perform better than the best individuals on letters-to-numbers problems: Effects of group size. *J. Pers. Soc. Psychol.* 90, 4 (2006), 644–651. DOI:https://doi.org/10.1037/0022-3514.90.4.644

[38] Patrick R. Laughlin, Norbert L. Kerr, James. H. Davis, Henry. M. Halff, and Kenneth. A. Marciniak. 1975. Group size, member ability, and social decision schemes on an intellective task. *J. Pers. Soc. Psychol.* 31, 3 (1975), 522–535. DOI:https://doi.org/10.1037/h0076474

[39] R. Lenth. 2018. Emmeans: Estimated marginal means, aka least-squares means. *R Package.*

[40] Russell V. Lenth. 2016. Least-squares means: the R package lsmeans. *J. Stat. Softw.* 69, 1 (2016), 1–33.

[41] Rivka Levitan, Agustin Gravano, Laura Willson, Stefan Benus, Julia Hirschberg, and Ani Nenkova. 2012. Acoustic-prosodic entrainment and social behavior. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (NAACL HLT '12), 11–19.

[42] Zachary C. Lipton, David C. Kale, and Randall Wetzel. 2016. Directly modeling missing data in sequences with RNNs: Improved classification of clinical time series. In *Proceedigns of Machine Learning Research*, 253–270.

[43] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the International Conference on Machine Learning.*

[44] Joseph H. Manson, Gregory A. Bryant, Matthew M. Gervais, and Michelle A. Kline. 2013. Convergence of speech rate in conversation predicts cooperation. *Evol. Hum. Behav.* 34, 6 (November 2013), 419–426. DOI:https://doi.org/10.1016/j.evolhumbehav.2013.08.001

[45] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan Yuille. 2015. Deep captioning with multimodal recurrent neural networks (M-RNN). In *Proceedings of the 2015 International Conference on Learning Representations.*

[46] Bernard A. Nijstad, Wolfgang Stroebe, and Hein F.M. Lodewijkx. 2003.

Production blocking and idea generation: Does blocking interfere with cognitive processes? *J. Exp. Soc. Psychol.* 39, 6 (November 2003), 531–548. DOI:https://doi.org/10.1016/S0022-1031(03)00040-4

[47] Hai X. Pham, Samuel Cheung, and Vladamir Pavlovic. 2017. Speech-driven 3D facial animation with implicit emotional awareness: A deep learning approach. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2328–2336. DOI:https://doi.org/10.1109/CVPRW.2017.287

[48] Daniel C. Richardson and Rick Dale. 2005. Looking to understand: The coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. *Cogn. Sci.* 29, 6 (November 2005), 1045–1060. DOI:https://doi.org/10.1207/s15516709cog0000_29

[49] Daniel C. Richardson, Rick Dale, and Natasha Z. Kirkham. 2007. The art of conversation is coordination. *Psychol. Sci.* 18, 5 (May 2007), 407–413. DOI:https://doi.org/10.1111/j.1467-9280.2007.01914.x

[50] Michael A. Riley, Michael J. Richardson, Kevin Shockley, and Verónica C. Ramenzoni. 2011. Interpersonal synergies. *Front. Psychol.* 2, (2011), 38. DOI:https://doi.org/10.3389/fpsyg.2011.00038

[51] Emanuel A. Schegloff. 2000. Overlapping talk and the organization of turn-taking for conversation. *Lang. Soc.* 29, 1 (2000), 1–63.

[52] Julian Schulze and Stefan Krumm. 2017. The "virtual team player": A review and initial model of knowledge, skills, abilities, and other characteristics for virtual collaboration. *Organ. Psychol. Rev.* 7, 1 (February 2017), 66–95. DOI:https://doi.org/10.1177/2041386616675522

[53] David A. Sears and James Michael Reagin. 2013. Individual versus collaborative problem solving: Divergent outcomes depending on task complexity. *Instr. Sci.* 41, 6 (November 2013), 1153–1172. DOI:https://doi.org/10.1007/s11251-013-9271-8

[54] Kevin Shockley, Daniel C. Richardson, and Rick Dale. 2009. Conversation and Coordinative Structures. *Top. Cogn. Sci.* 1, 2 (April 2009), 305–319. DOI:https://doi.org/10.1111/j.1756-8765.2009.01021.x

[55] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1 (2014), 1929–1958.

[56] I. D. Steiner. 1972. Group processes and group productivity. *New York Acad.* (1972).

[57] Chen Sun, Valerie Shute, Angela E.B. Stewart, Jade Yonehiro, Nicholas Duran, and Sidney K. D'Mello. Toward a generalized competency model of collaborative problem solving. *Rev.*

[58] Jacqueline Kory Westlund, Sidney K. D'Mello, and Andrew M. Olney. 2015. Motion Tracker: Camera-based monitoring of bodily movements using motion silhouettes. *PLoS One* 10, 6 (June 2015), e0130293. DOI:https://doi.org/10.1371/journal.pone.0130293

[59] Stephen J. Zaccaro. 1991. Nonequivalent associations between forms of cohesiveness and group-related outcomes: evidence for multidimensionality. *J. Soc. Psychol.* 131, 3 (June 1991), 387–399. DOI:https://doi.org/10.1080/00224545.1991.9713865

[60] Code Studio. Retrieved April 1, 2018 from https://studio.code.org/s/mc/stage/1/puzzle/1

[61] IBM. Retrieved May 2, 2018 from https://www.ibm.com/watson/services/speech-to-text/

[62] Keras. Retrieved May 2, 2018 from https://github.com/keras-team/keras

[63] Convolutional neural networks for visual recognition. Retrieved from http://cs231n.github.io/neural-networks-1/

[64] Scikit Learn. Retrieved May 3, 2018 from https://github.com/scikit-learn/scikit-learn

[65] Convolutional neural networks for visual recognition. Retrieved May 3, 2018 from http://cs231n.github.io/neural-networks-2/

[66] 2015. *PISA 2015 Collaborative Problem Solving Framework.*