International Society of
the Learning Sciences

Simulating Collaborative Discourse Data

Zachari Swiecki, Monash University, zach.swiecki@monash.edu
Cody Marquart, Wisconsin Center for Education Research, cody.marquart@wisc.edu
Brendan Eagan, University of Wisconsin, Madison, beagan@wisc.edu

Abstract: We present a method for simulating collaborative discourse in terms of the patterns of codes present in the data. We argue that simulation methods have a successful history in other areas of social science, and that these methods can—and should—be adopted to study collaborative processes. We describe a novel simulation method that we have developed and provide evidence of the validity of this method by statistically comparing our observed data to simulated data. Furthermore, we show that using simulated data yields insights about collaborative processes that we would have missed if we had only examined the observed data.

Introduction

Collaboration has widely been recognized as an essential skill for success in the 21st century (Griffin, 2012), and research has shown that collaboration can be good for learning (e.g., Miyake & Kirschner, 2014). In turn, many researchers in the learning sciences (LS) who study computer-supported collaborative learning (CSCL) have focused on the kinds of processes that occur when people work together. To study these processes, researchers collect discourse data that represents interactions among individuals. While real data is the gold standard for conducting analyses of collaborative learning, in most cases, collecting it is difficult, time-consuming, and expensive. Moreover, any conclusions drawn from the data are limited either to the sample collected or to other samples that are sufficiently similar.

In this paper, we present a method for simulating collaborative discourse data. To be clear from the outset, our method does not simulate discourse *per se*. When LS and CSCL researchers analyze discourse, they often *segment* it—for example, by turn of talk—and *code* it for specific themes of interest. Subsequent analyses are done by modeling these codes in terms of their frequency, sequence, or interaction. Our method simulates the patterns of codes present in segments of discourse, and thus affords the kinds of analyses of discourse data that are common in the LS and CSCL communities.

We argue that simulation methods have a successful history in other areas of social science, and that these methods can—and should—be adopted to study collaborative processes. To demonstrate the utility of simulations, we describe a novel simulation method that we have developed. We provide evidence of its validity by statistically comparing our observed data to simulated data, and we show that using simulated data yields insights about collaborative processes that we would have missed if we had only examined the observed data.

Theory

The Limitations of Data

Researchers need data to conduct empirical studies of collaborative processes. Often, these data are based on observations of collaborative discourse—the actual things people say and do as they interact. However, discourse data can be difficult, time-consuming, and expensive to collect—ethics applications must be drafted and approved, participants recruited and compensated, recording equipment and software purchased, data cleaned and stored securely. And all of this needs to happen before the painstaking process of analysis and mean-making can begin.

Once researchers obtain data, analyze it, and draw conclusions, they may consider the extent to which those conclusions generalize. However, the rules of statistical inference suggest that the conclusions drawn from an analysis using, say, a regression model, only generalize to data that are sufficiently similar to the sample on which the model was fit. In many cases, the generalization problem is not particularly salient. All it means is that researchers need to be more careful about the extent of the claims they make. However, when data is scarce and sample sizes are small, researchers may not be able to claim that their conclusions generalize at all.

Statisticians have developed tools to address issues of sample size and generalization, such the nonparametric bootstrap (Fox & Weisberg, 2018), which generates a sampling distribution for a test statistic by repeatedly sampling from the data on hand. Still, methods like this only afford generalization to data that looks like the data we already have. To address the question of whether our findings generalize to data that looks quite different from what we have, we can look to simulations.

Simulation

In *Simulation for the Social Scientist*, Gilbert and Troitzsch (2005) describe simulation as a method similar to statistical modeling. In the latter, researchers have some real-world phenomenon, or target, that they want to understand. Their aim is to create a model of the target that is easier to study than the target itself. To do so, they collect data and develop a model (e.g., a set of regression equations) that abstracts salient features of the target. This model includes some parameters (e.g., beta coefficients) whose magnitudes are determined by fitting the model to the data on hand. Finally, they test whether the model generates predictions that are sufficiently similar to the collected data (e.g., using a coefficient of determination) and examine the significance and relative magnitude of the estimated parameters (e.g., using *p* values and measures effect size).

Simulation proceeds similarly except that the model may be in the form of an algorithm or computer program instead of a set of equations, and this model is used to generate *simulated* data rather than predictions from real data. If possible, the simulated data is compared to available real data to test how similar the two are and assess the validity of the simulation.

In the social sciences, there is rich history of applying simulations (specifically, agent-based models) to study human behavior (see Gilbert & Troitzsch, 2005). However, in LS and CSCL simulations are seldom used. A notable exception comes from Hutchins (1995). In *Cognition in the Wild*, Hutchins used simulated data to show that the cognitive properties of a group may differ from those of the individuals who constitute the group and that the consequences of this property depend on how the group distributes tasks among its members. Despite appearing in one of the seminal works in our field, simulations have not been widely adopted in LS and CSCL as methods for understanding collaboration. Instead, work has mainly focused on their applications as tools for teaching complex systems (e.g., Jacobson & Wilensky, 2006).

Studying Collaborative Processes

As Hutchins argues (1995), simulations "are both a kind of notation system that forces one to be explicit about the theoretical constructs that are claimed to participate in the production of the phenomena of interest and a dynamical tool for investigating a universe of possibilities," (pgs. 261-262). In LS and CSCL, collaborative processes are phenomena of interest. Researchers have focused on them because they help us to understand how collaborative learning takes place. By understanding these processes, we can manipulate them in beneficial ways.

A widespread example of this kind of manipulation is the *jigsaw* (Slavin, 2011). In the jigsaw approach, each student on a team is assigned a unique topic on which to become an "expert". Those students with the same topic across teams meet in separate groups for discussion. Afterward, they return to their original teams to relay what they have learned. One mechanism that makes the jigsaw successful is that it creates *informational interdependence* (DeChurch & Mesmer-Magnus, 2010) within the team—that is, it creates conditions in which individuals need to *share different information with one another*. Thus, two factors that participate in the production of collaborative processes in the jigsaw approach are the *dissimilarity* of information being shared and the level of *interactivity* among teammates.

It is possible to construct a real scenario in which we implement a jigsaw, calculate measures of dissimilarity and interactivity, and relate them to changes in collaborative processes. We could record and transcribe team conversations pre-jigsaw and post-jigsaw. Then, segment the data and code it for themes relevant to the particular learning domain. Finally, we could compare patterns in these codes between the jigsaw conditions to say something meaningful about the effect of the manipulation, and potentially uncover mechanisms to target through interventions, such as collaboration scripts. This approach, however, falls short of investigating the "universe of possibilities" Hutchins was describing. The reason being that we have collected data from only a small slice of this universe. Simulations let us explore more of this universe.

Research Questions

To demonstrate the utility of simulations for LS and CSCL, we describe a method for simulating the kinds of data researchers encounter when studying collaboration. As the basis for our simulations, we used collaborative discourse data collected from an online pedagogical experience that employed a jigsaw design. Using these data, we tested whether variation in dissimilarity and interactivity explained differences between pre- and post-jigsaw conditions. After validating the simulation, we used simulated data to test whether we could see similar results under a broader, yet still plausible, range of dissimilarity and interactivity values than observed in the real data. The following research questions guided this work (a) "Do dissimilarity and interactivity explain differences between these individuals in pre- and post-jigsaw conditions?"; (b) "Can we accurately simulate coded data from this context?"; (c) and "Do dissimilarity and interactivity explain differences between simulated individuals?".

Methods

Data

Our observed data was collected from a single implementation of the virtual internship *Rescushell* at a large university in the Midwestern United States. Virtual internships are online educational experiences where participants act as interns at a fictional company (Chesler et al., 2015). In *Rescushell*, participants intern at the company RescuTek and work together to design and test robotic exoskeletons for rescue workers.

The internship uses a jigsaw approach: each participant is assigned to one of five teams and each team is assigned a unique exoskeleton component on which to base their designs. After exploring the results of using that one component in combination with other kinds of design inputs, the teams are shuffled such that each new team contains at least one person familiar with a particular component. Here, the components are *actuators*, or the motors used to initiate and control the movement of the exoskeleton. Other inputs include power sources and the kind material used to build the device—e.g., steel or aluminum. Teams test different combinations of inputs and assess the performance of their designs in terms of outputs such as battery life, safety rating, and cost. They judge design performance in relation to standards for these outputs set by internal consultants within the company.

During the internship, participants interacted via the online system *WorkPro*, which includes an integrated email and chat messaging service. All chats were automatically logged by the system and constitute the observed discourse data we analyzed for this study. As the chats were logged, they were automatically segmented by the current internship activity and turn of talk. Here, a turn of talk is the text sent when a participant hit return or clicked "send" in the messaging window. Activities were distinct sets of tasks that teams undertook at pre-defined points in the internship. As the focus of our study was collaborative discourse, we only analyzed chats from activities that required collaboration, such as design meetings. In total, the observed data we analyzed includes 1870 turns of talk from 26 participants.

Coding

The data were coded for eight themes related to engineering design in the context of *Resucshell*. In particular, the codes: `Electric`, `Series.Elastic`, `Hydraulic`, `Pneumatic`, and `PAM` refer to the different actuators that could be included in the designs—i.e., the jigsaw topics. `Technical.Constraints` refers to other design inputs such as batteries; `Performance.Parameters` refers to the outputs used to evaluate the designs, such as cost; and `Client.and.Consultant.Requests` refers to the standards set by the company's internal consultants.

The codes were applied using automated classifiers that we developed with the *nCoder* web application (Marquart et al., 2020). The classifiers used regular expressions to identify the presence or absence of each code in a given turn of talk (1). We validated the classifiers by comparing their decisions to those of two raters who had achieved sufficient interrater reliability. All raters—two human raters and the classifiers—achieved Cohen's kappa > 0.90 and Shaffer's rho < 0.05 for these data.

Analysis

Epistemic Network Analysis

To analyze patterns of discourse in the observed data, we used the R implementation of *epistemic network analysis* (ENA)—`rENA` (Marquart et al., 2019) (2). For a given unit of analysis, ENA measures connections between codes that occur within segments of discourse defined by a moving window of fixed length. Connections are defined as the co-occurrence of any pair of codes within the window. This process results in an adjacency matrix for each unit of analysis, where the cells of the matrix are the number of times a given pair of codes co-occurred. The upper triangles of these matrices are converted to *adjacency vectors*, normalized, and projected into a low dimensional *ENA space* via singular value decomposition (3).

ENA represents the normalized adjacency vector for a given unit of analysis in two ways: as a point in the ENA space (an *ENA* score) and an un-directed and weighted network where nodes are the codes and edges are the values of the normalized adjacency vector. Any two networks can be compared by subtracting their edge weights to show the connections that are stronger in one network compared to the other. Both ENA scores and networks can be aggregated to visualize the average connections made by sub-populations within the data.

We defined sub-populations according to the jigsaw conditions. Therefore, we considered each participant as two separate units of analysis, one pre-jigsaw and one post-jigsaw. We chose a window size of two turns of talk (4). To highlight the differences between the two conditions, the ENA projection method maximized the variance accounted for between these two groups on the first dimension of the space.

Mixed-Effects Modeling

We tested effect of dissimilarity and interactivity on connection patterns using mixed-effects models. To operationalize dissimilarity, for each unit, we calculated an adjacency matrix whose cells contained the probability at which that co-occurrence was observed in their data (5). For a given unit, we calculated the average Euclidean distance between their matrix and their teammates matrices to obtain the dissimilarity metric. Larger values suggested that the individual made connections that were dissimilar to their teammates; smaller values suggested that they made connections that were similar to their teammates.

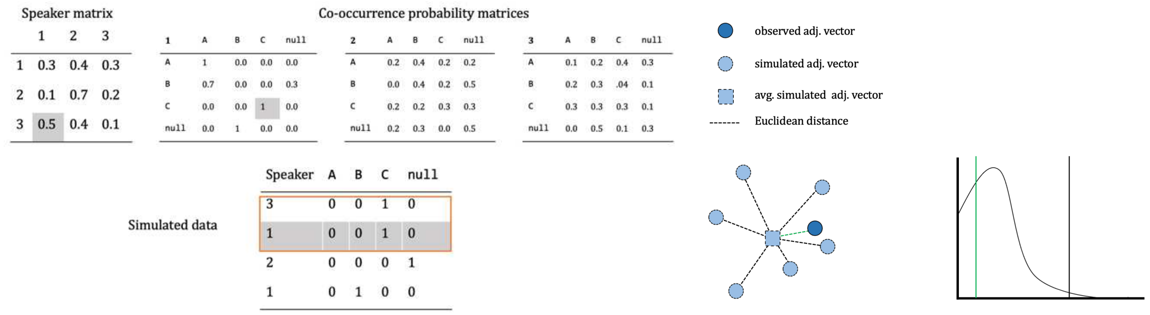
To operationalize interactivity, for each team, we first calculated the lag-1 transition matrix of their turns of talk. For a given team, the cells of this matrix represent the probability that an individual on the column had turns that came directly after the individual on the row. On the diagonal is the probability that an individual had turns that directly followed their own. We subtracted these diagonal values from 1 to obtain the interactivity metric. Higher values meant that the individual tended to respond to their teammates' turns of talk; lower values meant that they tended to respond to their own turns.

The observed data has a nested structure: units of analysis are nested within individuals, who are in turn nested within teams. To account for nesting, we used the mixed-effects modeling via the R package `lmerTest` (Kuznetsova et al., 2017). The outcome variable was the first dimension ENA score for each unit. We modeled dissimilarity and interactivity as fixed effects and a team indicator as a random effect. The regression equations also included fixed effects for the group means of the dissimilarity and interactivity metrics to distinguish within-group and between-group effects. We tested multiple models with different combinations of predictors and selected the best model based on the presence of significant predictors and lowest AIC value.

Simulation Method

The key components of the simulation are the cognitive and social *generating matrices*. In form, these are the same as the matrices described above that we used to operationalize dissimilarity and interactivity. For the simulation, however, these matrices were generated by a pseudo-random process rather than calculated directly from the observed data.

Figure 1.
Overview of simulation method (a) and simulation validation metric (b)



Speaker matrix

	1	2	3
1	0.3	0.4	0.3
2	0.1	0.7	0.2
3	0.5	0.4	0.1

Co-occurrence probability matrices

1	A	B	C	null
A	1	0.0	0.0	0.0
B	0.7	0.0	0.0	0.3
C	0.0	0.0	1	0.0
null	0.0	1	0.0	0.0

2	A	B	C	null
A	0.2	0.4	0.2	0.2
B	0.0	0.4	0.2	0.5
C	0.2	0.2	0.3	0.3
null	0.2	0.3	0.0	0.5

3	A	B	C	null
A	0.1	0.2	0.4	0.3
B	0.2	0.3	.04	0.1
C	0.3	0.3	0.3	0.1
null	0.0	0.5	0.1	0.3

Simulated data

Speaker	A	B	C	null
3	0	0	1	0
1	0	0	1	0
2	0	0	0	1
1	0	1	0	0

(a)

(b)

For example, say that we want to simulate data for a team of three individuals whose discourse is coded for three codes. First, we generate one speaker transition matrix for the team. To generate the values for this matrix, we randomly select values from 0 to 1 and normalize the rows such that they sum to 1. Next, we generate a co-occurrence probability matrix for each individual on the team. This matrix is generated such that all rows sum to either one or zero. Here, a row summing to zero indicates that the simulated individual will never have a window within which the code represented by that row appears. To ensure that the possible co-occurrence probabilities reflect those that we might expect to see in the real data, we randomly sample from the distribution of possible values for that co-occurrence found in the observed data.

Given the set of co-occurrence probability matrices and the speaker transition matrix for the team, the simulation process proceeds as follows (see Figure 1a). ***Generate the sequence of the speakers using the speaker transition matrix*:** Randomly select the first speaker. Select the subsequent speaker based on the probabilities in the current speaker's row of the transition matrix. Repeat until the desired number of turns of talk for the simulated team is reached. ***For each speaker in the sequence, use their co-occurrence probability matrix to generate a binary code vector for their turn of "talk"*:** For the first speaker, randomly select the present code in the vector based on the possible codes for that speaker—i.e., non-zero rows of their matrix. Set all other codes in the vector to zero. Select the present code for the subsequent speaker based on their matrix and the code vector of the previous turn—i.e., whichever code is present in the previous turn, the algorithm finds that corresponding row in subsequent speaker's matrix and selects their present code based on the probabilities in that row. Set all other codes in the vector to zero. Repeat until all speakers in the sequence have a code vector.

Simulation Validation

To validate the simulation method, we compared the observed and simulated data in terms of the patterns of codes in each. We simulated data using initial conditions obtained from the observed data: the number of teams (ten); the number individuals on each team (five to six); the number of turns each individual produced; the number of codes applied to the data (eight); the sets of co-occurrence probability matrices for each team; and the speaker transition matrices for each team. Thus, for each team, the simulation method was used to create the same number of coded turns of talk observed in the real data.

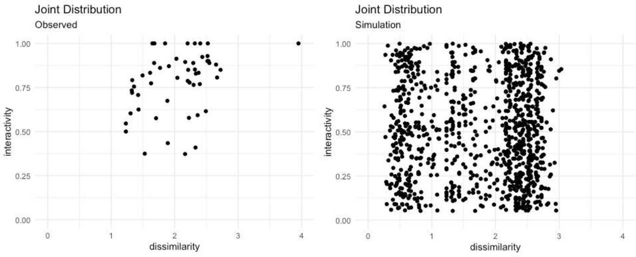
The extent to which these simulated turns contain code patterns that are similar to the patterns in the observed data (as measured by ENA), evaluates the validity of the simulation. Due to the stochastic nature of the simulation method, each teams' data was simulated 1000 times and ENA models (with the same specifications described above) were used to generate adjacency vectors for each unit in each run.

We employed the method developed by Swiecki (2021) (Figure 1b) to test whether the observed and simulated adjacency vectors were part of the same distribution. This method calculates the distribution of the average Euclidean distances between the simulated adjacency vectors and their means. This is a distribution of similarity measures between adjacency vectors under the null hypothesis that they come from the same distribution. Then, the method calculates the average Euclidean distance of the observed adjacency vectors to their corresponding mean simulated vectors. Finally, we compare that average (our test statistic) to the values in the null hypothesis distribution. If the statistic is less than the 95% percentile of the null hypothesis distribution, the test suggests that the observed and simulated vectors are from the same distribution. In other words, failing to reject the null hypothesis suggests that the observed and simulated vectors are very similar, and thus, that the data on which these vectors were generated is very similar.

Analysis of Simulated Data

The observed data used to fit the mixed-effects model described above contains a narrow range of the possible values for dissimilarity and interactivity (see Figure 2a). The simulation method allowed us to test the relationships among dissimilarity, interactivity, and connection patterns for a greater range of those metrics (see Figure 2b).

Figure 2
Joint distribution of interactivity and dissimilarity for observed (a) and simulated (b) data



(a) (b)

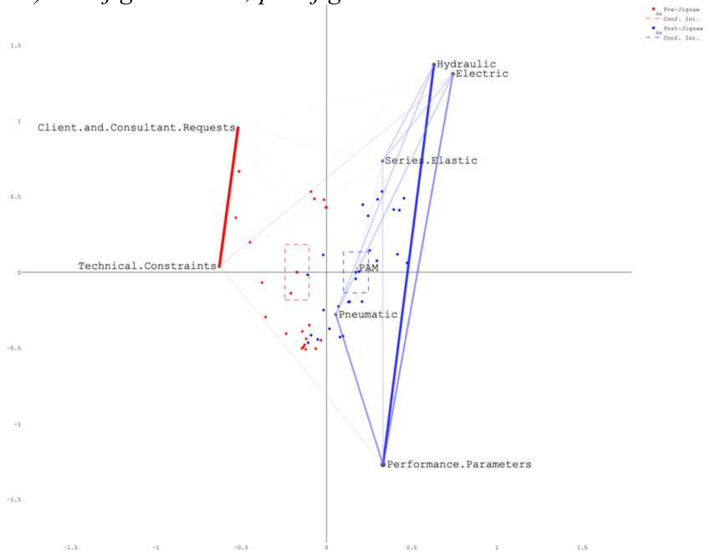
We ran the simulation 1000 times (6) with the following parameters: 200 teams; 5 individuals per team; 187 turns of talk per team (7); and 8 codes applied to the data. For each run, we calculated dissimilarity and interactivity metrics (based on the generating matrices) for each individual and modeled the data using the ENA space produced from the observed data (8). We then ran a mixed-effects model on the data that regressed the first dimension ENA scores on fixed effects for dissimilarity and interactivity, their means, an interaction term, and a random effect for team (9). We averaged the regression coefficients across the simulation runs and calculated *bias corrected, accelerated* (BC_a) *percentile intervals* (Fox & Weisberg, 2018) to test their significance.

Results

Observed Data

Figure 3 shows the ENA space obtained from the observed data. The dimensions of the space can be interpreted via the connections at the extremes. Here, we focus on the first dimension which distinguishes individuals who made connections between `Client.and.Consultant.Requests` and `Technical.Constraints` versus those who made connections among the actuator codes (jigsaw topics) and `Performance.Parameters`. The network subtraction shows that individuals in the pre-jigsaw condition (red) made relatively more connections among the codes on the left, while in the post-jigsaw condition (blue) made relatively more connections among the codes on the right. This result aligns with the intended purpose of the jigsaw design. Pre-jigsaw teams only had access to information about one kind of actuator; post-jigsaw teams had access to information about all five actuators and could thus make connections among them in their discourse.

Figure 3
ENA space (observed data). Pre-jigsaw in red; post-jigsaw in blue. Points are ENA Scores. Squares are means.



The best regression model for the observed data is shown in tables 1 and 2. The results show that the mean dissimilarity metric is significantly associated with the first dimension ENA score, controlling for team effects. One unit increase in mean dissimilarity is associated with an average increase in the ENA score by 0.34 units. In other words, individuals on teams that shared different kinds of information tended to talk more like post-jigsaw teams. Given that the outcome measure ranges from -0.47 to 0.53, the effect size of this association (indicated by the magnitude of the regression coefficient) suggests that the effect is relatively large.

Table 1
Regression results (observed)

	ENA 1		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	-0.69	-1.12 – -0.26	**0.002**
dissimilarity (mean)	0.34	0.13 – 0.54	**0.002**

Table 2
Regression results, random effects (observed)

σ^2	0.03
$\tau_{00\ \text{Team}}$	0.02
ICC	0.37

Simulation Validation

The validation test produced a distribution of adjacency vectors for each simulated participant based on initial conditions from the observed data. The 95th BC_a percentile value for the null hypothesis distribution of average distances derived from these vectors was 0.30. The test statistic was 0.09, suggesting that, on average, the observed adjacency vectors were statically indistinguishable from the simulated adjacency vectors. In other words, the patterns of connections modeled in the observed and simulated data are indistinguishable. Thus, we have evidence that the observed data and simulated data (on which these models were applied) are also indistinguishable for the purposes of this analysis.

Simulated Data

The results of the simulation study are shown in Table 3. The first row shows the average regression coefficients across the 1000 simulation runs. The second and third rows show the lower and upper bounds of the 95% BC_a

percentile intervals. We can interpret coefficients with intervals that include zero as statistically insignificant. The results show that all of the model terms except the interaction term are significant. The range of the outcome variable across the simulation was [-0.76 - 0.55], which suggests that the effect sizes of the significant associations are small to moderate. At the individual level, as dissimilarity increases, patterns of connections look more like pre-jigsaw participants; and as interactivity increases, patterns of connections look more like pre-jigsaw participants. At the team level, as dissimilarity increases, patterns of connections look more like post-jigsaw participants; and as interactivity increases, patterns of connections look more like post-jigsaw participants.

Table 3
Regression results (simulation)

	Intercept	Dissimilarity	Interactivity	Diss.mean	Int.mean	Diss*Int
Mean.estimate	**-0.08**	**-0.11**	**-0.11**	**0.14**	**0.12**	0.01
Lower.ci	-0.14	-0.18	-0.2	0.07	0.02	-0.03
Upper.ci	-0.02	-0.03	-0.03	0.21	0.22	0.05

Discussion

Together, the results show that (a) dissimilarity of talk at the team level is significantly associated with discourse patterns in this implementation of *Rescushell*; (b) the simulation method can produce coded data that is statistically similar to the observed coded data; and (c) both dissimilarity and interactivity are significantly associated with discourse patterns in the simulated data at the individual and team levels.

These results suggest that increasing the dissimilarity of information shared and the interactivity among teammates—perhaps through collaboration scripts—can increase the effect of the jigsaw. More importantly, though, the results show the limitations of observed data and the potential affordances of simulated data. Because of the sample of real data that we happened to have, we were forced to conclude that only dissimilarity (at the team level) had a significant effect. Simulating data allowed us to go beyond the sample we happened to have and test the relationships between dissimilarity, interactivity, and discourse patterns under a greater range of conditions. Because of the simulation, we were able to draw conclusions that we otherwise would have missed. We were able to draw these conclusions without undertaking the arduous process of collecting additional data.

This study has several important limitations. First, as mentioned in the introduction to the paper, the method does not simulate discourse; it simulates the patterns of codes that researchers might apply to the discourse. As such, it does not afford many of the more nuanced analyses that might be applied to discourse, such as conversation analyses that focus on word frequencies, pauses, intonations, and so forth. However, because it simulates the codes applied to discourse, the method is very flexible. In principle, if the phenomena can be coded for and a plausible generating mechanism can be defined, simulation can proceed.

Second, our simulation method is constrained to data that is mutually exclusively coded (one code per turn of talk) and to relationships between individuals and codes that occur within a window of two turns. Data from many observed scenarios may warrant analyses that exceed these constraints. However, the purpose of this study was not the particulars of the data, the codes we applied, or the models we used; our purpose was to provide a plausible argument for the utility of simulated data in the context of studying collaboration. Of course, the particulars of the data, codes, and models will be important for deriving useful results from simulations like this one in the future. As such, we will continue to explore ways to expand the method beyond its current constraints. For example, it should be possible to increase the size of the window within which interactions are simulated by using higher-order Markov models instead of lag-1 transition matrices.

Finally, this simulation only manipulated two discourse parameters—dissimilarity and interactivity. This choice was a conscious one on our part to simplify the method and argument for the study. In principle, other potentially important parameters could be operationalized and included in the simulation, such as the formation of social cliques within teams. Our future work will develop methods to incorporate a greater variety of parameters into the simulation framework.

Conclusion

We presented a method for simulating coded discourse data from collaborative scenarios. We showed that our method is able to produce coded data that is statistically similar to observed data. Furthermore, we showed that by modeling the simulated data we were able to draw conclusions that we missed when only examining the observed data. Observed data will always be the gold standard, and we do not mean to suggest that simulated data should replace it. However, our results suggest that simulation can be a useful approach for understanding collaborative processes that occur in plausible, yet unobserved, conditions.

Endnotes

(1) Our automated classification scheme could assign more than one code to a given turn of talk. However, our simulation method was limited to generating data in which turns were mutually exclusively coded. To accommodate this, for turns of talk that were coded for multiple codes, we randomly selected one of the codes to apply exclusively.

(2) R code for the analysis can be found at: https://github.com/zlswiecki/cscl_2022_swiecki_marquart_eagan.

(3) A detailed description of ENA can be found in Bowman and colleagues (2021).

(4) Prior work (Ruis et al., 2019) suggests that the optimal window size for these data is seven turns of talk. However, alignment with the simulation method required that we use a window of two turns.

(5) Unlike the adjacency matrices calculated by ENA, these matrices included the diagonal–i.e., the probabilities of a code co-occurring with itself—as well as a row and column for the absence of any codes, termed "null".

(6) Increasing the number of simulation runs yielded similar results.

(7) This was the average value obtained from the observed data.

(8) That is, we projected the adjacency vectors from the simulation into the ENA space of the observed data using the rotation matrix obtained from the dimensional reduction on the observed data.

(9) Preliminary tests of the simulation suggested that the interaction between the mean metrics was not significant. Therefore, that term was excluded from the final simulation model. Three-way interactions were excluded to ease interpretation.

References

Bowman, D., Swiecki, Z., Cai, Z., Wang, Y., Eagan, B., Linderoth, J., & Shaffer, D. W. (2021). The mathematical foundations of epistemic network analysis. In A. R. Ruis & S. B. Lee (Eds.), *Advances in Quantitative Ethnography* (pp. 91–105). Springer International Publishing.

Chesler, N. C., Ruis, A. R., Collier, W., Swiecki, Z., Arastoopour, G., & Williamson Shaffer, D. (2015). A novel paradigm for engineering education: Virtual internships with individualized mentoring and assessment of engineering thinking. *Journal of Biomechanical Engineering*, *137*(2).

DeChurch, L. A., & Mesmer-Magnus, J. R. (2010). The cognitive underpinnings of effective teamwork: A meta-analysis. *Journal of Applied Psychology*, *95*(1), 32–53.

Fox, J., & Weisberg, S. (2018). *An R companion to applied regression*. Sage Publications.

Gilbert, N., & Troitzsch, K. (2005). *Simulation for the social scientist*. McGraw-Hill Education.

Griffin, P. (2012). *Assessment and teaching of 21st century skills*. Springer.

Hutchins, E. (1995). *Cognition in the wild*. MIT Press.

Jacobson, M. J., & Wilensky, U. (2006). Complex systems in education: Scientific and educational importance and implications for the learning sciences. *Journal of the Learning Sciences*, *15*(1), 11–34.

Kuznetsova, A., Brockhoff, P. B., & Christensen, R. H. B. (2017). lmerTest package: Tests in linear mixed effects models. *Journal of Statistical Software*, *82*(13).

Marquart, C. L., Hinojosa, C. L., Eagan, B. R., Siebert-Evenstone, A. L., & Shaffer, D. W. (2020). *NCoder [Online software]* (0.2.2.0) [Computer software]. http://app.n-coder.org

Marquart, C. L., Swiecki, Z., Collier, W., Eagan, B. R., Woodward, R., & Shaffer, D. W. (2019). *RENA: Epistemic network analysis [R package]* (0.1.6.1) [Computer software]. https://cran.r-project.org/web/packages/rENA/index.html

Miyake, N., & Kirschner, P. A. (2014). The social and interactive dimensions of collaborative learning. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences* (pp. 418–438). Cambridge University Press.

Ruis, A. R., Siebert-Evenstone, A. L., Pozen, R., Eagan, B. R., & Shaffer, D. W. (2019). Finding common ground: A method for measuring recent temporal context in analyses of complex, collaborative thinking. *13th International Conference on Computer Supported Collaborative Learning (CSCL)*, *1*, 136–143.

Slavin, R. E. (2011). Instruction based on cooperative learning. *Handbook of Research on Learning and Instruction*, *4*.

Swiecki, Z. (2021). The expected value test: A new statistical warrant for theoretical saturation. *Advances in Quantitative Ethnography: Third International Conference, ICQE 2021, Malibu, CA, USA, November 6-11, Proceedings*. International Conference on Quantitative Ethnography 2021.

Acknowledgments

This work was funded in part by the National Science Foundation (DRL-1661036, DRL-1713110, DRL-2100320), the Wisconsin Alumni Research Foundation, and the Office of the Vice Chancellor for Research and Graduate Education at the University of Wisconsin-Madison. The opinions, findings, and conclusions do not reflect the views of the funding agencies, cooperating institutions, or other individuals.