Do Speech-Based Collaboration Analytics Generalize Across Task Contexts?

Samuel L. Pugh
University of Colorado Boulder
samuel.pugh@colorado.edu

Arjun R. Rao
University of Colorado Boulder
arjun.rao@colorado.edu

Angela E.B. Stewart
Carnegie Mellon University
angelast@andrew.cmu.edu

Sidney K. D'Mello
University of Colorado Boulder
sidney.dmello@colorado.edu

ABSTRACT

We investigated the generalizability of language-based analytics models across two collaborative problem solving (CPS) tasks: an educational physics game and a block programming challenge. We analyzed a dataset of 95 triads (N=285) who used videoconferencing to collaborate on both tasks for an hour. We trained supervised natural language processing classifiers on automatic speech recognition transcripts to predict the human-coded CPS facets (skills) of constructing shared knowledge, negotiation / coordination, and maintaining team function. We tested three methods for representing collaborative discourse: (1) deep transfer learning (using BERT), (2) n-grams (counts of words/phrases), and (3) word categories (using the Linguistic Inquiry Word Count [LIWC] dictionary). We found that the BERT and LIWC methods generalized across tasks with only a small degradation in performance (Transfer Ratio of .93 with 1 indicating perfect transfer), while the n-grams had limited generalizability (Transfer Ratio of .86), suggesting overfitting to task-specific language. We discuss the implications of our findings for deploying language-based collaboration analytics in authentic educational environments.

CCS CONCEPTS

• **Human-centered computing** → Collaborative and social computing; • **Applied computing** → Education.

KEYWORDS

Natural language processing, Collaboration analytics, Collaborative problem solving

ACM Reference Format:
Samuel L. Pugh, Arjun R. Rao, Angela E.B. Stewart, and Sidney K. D'Mello. 2022. Do Speech-Based Collaboration Analytics Generalize Across Task Contexts?. In *LAK22: 12th International Learning Analytics and Knowledge Conference (LAK22), March 21–25, 2022, Online, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3506860.3506894

LAK22, March 21–25, 2022, Online, USA

ACM ISBN 978-1-4503-9573-1/22/03.
https://doi.org/10.1145/3506860.3506894

1 INTRODUCTION

Collaborative problem solving (CPS) - defined as two or more people engaging in a coordinated attempt to construct and maintain a joint solution to a problem [39] - is considered a critical skill for the 21st century workforce [16, 20]. In the modern knowledge economy, there is an increasing demand for workers capable of collaborating with diverse teams, effectively sharing their knowledge and skills, and communicating across disciplines to solve complex scientific and societal problems. However, the 2015 Programme for International Student Assessment (PISA) assessment found significant deficiencies in student's CPS competencies, with less than 10% of all students achieving the highest level of proficiency [32]. Accordingly, educational researchers, practitioners, and policymakers across the globe have emphasized the importance of improving and expanding CPS instruction in our education systems [10, 16, 17, 20, 31].

One shortcoming of CPS education is the lack of consistent assessment and diagnostic feedback [16]. Although students regularly engage in group work in schools, with most classes instituting some form of collaborative assignment, students rarely receive meaningful instruction or feedback on collaboration itself. It is widely acknowledged, however, that feedback is crucial to improving knowledge and skill acquisition [46]. This deficiency in our education systems represents a valuable opportunity for Learning Analytics solutions to improve CPS assessment and instruction.

Collaboration analytics refers to the techniques and approaches used to automatically (or semiautomatically) capture, analyze, mine, and distill data about collaborators' interactions [30, 41]. The content of communications is a particularly important stream of data when analyzing CPS interactions in an open-ended, human-to-human setting [20], and has been shown to provide evidence of CPS competence [1]. Previous work has applied natural language processing (NLP) techniques to communications between team members in order to automatically assess CPS [18, 23, 36, 51]. In this approach, open-ended communications between team members (either transcribed from speech or gathered from text chats) are analyzed by NLP algorithms, which are trained to detect indicators of CPS. These language-based models have been shown to accurately detect CPS skills [18, 23], generalize to out-of-sample teams [36, 51], and be robust to speech recognition errors [51], even when using data gathered in noisy school environments [36].

However, a major gap in this research is the assessment of these models' generalizability to different task contexts. The previous studies either used data from a single CPS task [18, 23, 51], or combined data from two CPS tasks [36]. Thus, it has not yet been

explored whether models trained on language from one task context (source) will generalize to a different task context (target), without providing additional labeled (or unlabeled) data from the target context. This represents a considerable barrier to deploying such models in real-world educational settings, where collaborative activities are not confined to a single task context, and pre-existing data from a new task (i.e., to perform domain adaptation techniques [37]) may not be available.

We address this limitation by investigating the degree to which speech-based models of CPS generalize across tasks. Specifically, we examine two CPS tasks: (1) Physics Playground (an educational physics game) and (2) Minecraft Hour of Code (a block programming challenge). We train supervised NLP models to predict the three CPS facets of (1) constructing shared knowledge, (2) negotiation / coordination, and (3) maintaining team function, derived from an empirically validated CPS competence model [56]. We assess generalizability by comparing the accuracy of models that are trained and tested on data from the same task (e.g., train on Physics data, test on Physics data) with models trained on data from one task and tested on data from the other task (e.g., train on Minecraft data, test on Physics data).

We explore this question by conducting a systematic comparison of three NLP models, each of which utilizes a different method for feature representation with unique theoretical groundings: (1) a deep transfer learning approach, (2) an open-vocabulary n-gram approach, and (3) a dictionary-based approach. While these three methods of representing language may differ in terms of accuracy, generalizability, transparency, bias/fairness, or required data quantity, in this paper we focus on assessing generalizability across task contexts. In doing so, we take an important step towards the goal of developing collaborative analytics models capable of supporting CPS assessment and instruction in authentic educational environments.

2 RELATED WORK

Our work is grounded in extensive research on linguistic modelling of CPS. Although non-verbal modalities (e.g., facial expressions, eye gaze, body movements, log files) have also been explored to model CPS, here we focus on studies that leveraged speech or language data. Previous research has used low-level features derived from speech signals (e.g., speech rate, acoustic-prosodic features) to predict CPS outcomes such as task performance [54, 58], group rapport [29], and active participation [53]. Often these low-level, non-semantic representations of speech are combined with other modalities in a multimodal learning analytics approach (e.g., [58]). However, extensive research has also used more advanced NLP methods to analyze CPS language at the semantic and syntactic levels. For example, linguistic data (either gathered from text chats or transcribed from speech) has been used to model important CPS skills such as negotiation [18, 23, 36, 51, 52], information sharing [18, 23, 36, 51], regulation [15] and argumentation [6, 40], in addition to predicting CPS outcomes such as learning gains [38, 47] and task performance [7, 9].

A popular NLP approach in these studies involves using counts of words or phrases (n-grams) as features for a classifier [18, 23, 36, 40, 51], although researchers have also tested additional lexical features such as punctuation [18] or part-of-speech tags [15, 55]. Recently, pre-trained deep neural networks have been increasingly used for NLP tasks, and several studies have demonstrated their efficacy in analyzing collaborative discourse [28, 36]. Yet another approach involves utilizing pre-existing NLP tools (e.g., Coh-Metrix [21]) to generate features indexing cohesion, linguistic alignment, lexical sophistication, or syntactic complexity. This approach has been applied to CPS discourse to identify emergent sociocognitive roles [13], model intra- and interpersonal dynamics [12], identify creativity [48], as well as predict task performance and learning outcomes [9, 38, 47].

Although linguistic modelling of CPS is an active area of research, little work has investigated the generalizability of language-based models to different educational task contexts. However, researchers have evaluated model generalizability across tasks in other domains. For example, Sharma et al. [44] trained models to predict cognitive performance (e.g., skill acquisition, problem solving) using physiological measures and facial expressions, and provided evidence of task-generalizability (i.e., across gaming, coding tasks) using engineered features. Similarly, [14, 49] investigated whether mind wandering detection models generalized across different task contexts (e.g., reading a text, viewing a film) using facial expressions [49] or eye movements [14]. Both found that the models generalized across some (but not all) tasks, and that careful feature engineering was necessary to achieve generalizability. Baker et al. [2] explored the generalizability of models that detect students "gaming the system" when interacting with an intelligent tutoring system. They found that the models generalized across different lessons (e.g., geometry, probability) with only a small degradation in performance. Similarly, Hutt et al. [24] investigated affect detection models in an online math learning platform, and demonstrated that models using generic (i.e., platform-agnostic) interaction features generalized across curricula (e.g., algebra, geometry).

We identified only one study [34] which examined the generalizability of linguistic features (i.e., NLP models) across educational contexts. In this work, Patikorn et al. trained NLP models (using a bag-of-words approach) to predict the knowledge components needed to solve math problems. They achieved high within-sample accuracy (using cross-validation on their dataset) but found that the models did not generalize to another sample of math problems, where performance degraded to near chance.

3 CONTRIBUTION AND NOVELTY OF CURRENT STUDY

A major deficiency in the literature on language-based collaboration analytics is the evaluation of model generalizability across different task contexts. To our knowledge, this is the first study to examine the generalizability of NLP-based collaborative analytics models between two distinct tasks. Our goal is to examine how different ways of modeling collaborative language navigate the accuracy vs. generalizability tradeoff. We do this by comparing three different feature representations used to model CPS discourse: (1) a state-of-the-art deep transfer learning approach (using the Bidirectional Encoder Representations from Transformers or BERT model [11]), (2) an open vocabulary n-gram approach [43], and (3) a dictionary-based approach (using the Linguistic Inquiry Word Count or LIWC

[57]). As elaborated below, these methods differ both in terms of their ability to capture the contextual semantics of language as well as the amount of world knowledge encoded in their representations. For example, BERT is able to learn sophisticated semantic representations of language through its multi-layer bidirectional Transformer architecture, and it acquires considerable linguistic knowledge through its extensive unsupervised pre-training (e.g., on all of English Wikipedia). In contrast, the more naïve n-gram representation simply encodes counts of words and phrases, offering little ability to represent their broader context or meaning. Finally, the LIWC approach leverages human knowledge to encode the psychological meaning of words using theoretically grounded and psychometrically-validated dictionaries.

We explore the advantages and disadvantages of each approach by comparing performance in terms of within-task accuracy and across-task generalizability. We hypothesize the following patterns for across-task generalizability: *BERT* > *LIWC* > *N-Grams* (since BERT encodes substantial linguistic knowledge whereas n-grams directly encode task-specific language) and within-task accuracy: *BERT* > *N-Grams* > *LIWC* (since LIWC uses a restricted set of features). We also conduct two auxiliary experiments to uncover empirical differences in these representations and their ability to classify collaborative language. This work serves to inform model selection decisions in future research on language-based collaboration analytics. It does not, however, aim to address deficiencies in models that fail to generalize across tasks, an item for future work.

4 DATASET

The dataset was collected for a previous project on remote CPS [50]; only pertinent details are discussed here.

4.1 Participants

The study involved 288 students (average age of 22 years) from two large public universities in the Western United States (111 from School 1 and 177 from School 2). 54% self-reported as female, 41% as male, 1% as non-binary/third gender, and 4% did not report gender. Participants self-reported race: 48% Caucasian, 25% Hispanic/Latino, 17% Asian, 3% Black or African American, 1% American Indian or Alaska Native, 3% Other, and 3% did not report. Participants were assigned to 96 triads based on scheduling constraints. Forty-six participants from 25 teams indicated they knew at least one person from their team prior to participation. The participants were compensated with a $50 Amazon gift card (95.8%) or with course credit (4.2%).

4.2 CPS Tasks

Our dataset contains two distinct CPS tasks. Physics Playground [45] is an educational game designed for learning of basic physics concepts (e.g., Newton's laws, energy transfer, and properties of torque) through gameplay. The goal of the game is to draw physics objects (e.g., ramps, levers, pendulums, springboards) in order to guide a ball to hit a balloon target. All objects in the game, both pre-existing in the level and those drawn by participants, obey the laws of physics (Figure 1A).

The Minecraft-themed Hour-of-Code environment [8] uses block-based programming to teach basic programming concepts in an interactive manner. Programming constructs (e.g., if-else statements) are represented as interconnecting blocks that fit together if the resulting code is syntactically correct. The assembled blocks control the actions of a Minecraft game character (who can move around, destroy blocks, place blocks etc.), and users can run and preview the results of their code in real time (Figure 1B).

4.3 Procedure

The study consisted of a brief at-home portion, followed by an in-lab session. For the at-home portion, each participant first completed a Qualtrics survey designed to assess individual difference measures. After the survey, participants completed a tutorial on how to use the Physics Playground and Minecraft environments.

During the in-lab session, participants were assigned to one of three computer-enabled workstations, either in separate rooms or partitioned in the same room using dividers (depending on the university). All interactions between participants took place via Zoom (https://zoom.us), a video conferencing platform with recording and screen-sharing capabilities. The shared screen (see Figure 1) and separate audio streams were recorded for each participant. Other data streams that are not analyzed in the current study were also recorded.

In the study, teams collaborated in a series of four 15-minute blocks. During the first three blocks, teams attempted to solve a series of game levels in Physics Playground. Then, in the fourth block, teams completed a CPS task using the Minecraft environment. The objective of this task was to use the code blocks to write a program satisfying five design requirements: 1) Build a 4x4 brick building; 2) Build at least 3 bricks of the building on water; 3) Use at least one if statement; 4) Use at least one repeat loop; 5) Use 15 blocks of code or less. The order of tasks was fixed as the goal of the original study was to investigate transfer of CPS skills from Physics Playground to Minecraft.

In each 15 minute block, one randomly-assigned team member was selected to control interaction with the CPS environment. This was done to facilitate collaboration rather than individual work, and because the interfaces only allow one controller at a time. The controller's screen was shared with the observers via Zoom screen-share, and the other two teammates could contribute to the solution however they saw fit. The role of controller was rotated between the team members for each block, such that each student was given a chance to be the controller for at least one block.

5 DATA PROCESSING

5.1 Data Exclusion

Transcripts from several teams were not available due to technical issues or insufficient time to complete the study. Out of 96 total teams, we analyzed data from 95 teams. Of these, 94 teams had available Physics transcripts and 88 teams had available Minecraft transcripts (87 teams had both).

5.2 Automatic Speech Transcription

We used the IBM Watson's Speech to Text service [25] to generate transcripts of the audio files recorded from each participant. The service generated start and stop times for each utterance. These

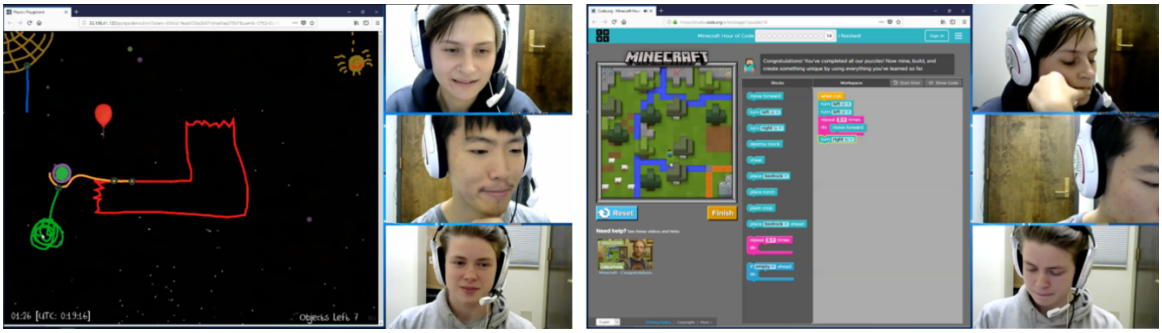


Figure 1: (A) Participants in a triad use videoconferencing to collaborate as they solve a level in the Physics Playground environment. (B) A team assembles code blocks in the Minecraft environment to control their Minecraft character.

timestamps were used to interleave the transcripts from each participant to produce a team-level transcript for each 15-minute collaboration block. When an utterance was incorrectly split into multiple segments, we combined sequential utterances if they belonged to the same participant and were less than two seconds apart (selected based on an analysis of different segmentation thresholds). To assess transcription accuracy, we manually transcribed a sample of 350 utterances from both tasks and computed a word error rate (WER) for each utterance, defined as (substitutions + insertions + deletions) / (words in human transcript). The mean WER was 32.8% for the Physics sample, and 30.5% for the Minecraft sample, indicating similar transcription accuracy across the two tasks. We did not investigate the effect of transcription errors in this work, however, previous studies [36, 51] have demonstrated that CPS constructs can be accurately modeled from automatic transcriptions of similar quality.

5.3 Human Coding of CPS Facets

We enlisted trained human experts to annotate each automatically transcribed utterance using our validated CPS framework [56]. The framework was developed based on a synthesis of four extant frameworks and was intended to generalize across task domains, which made it ideal for the present case. The framework defines three core CPS facets: (1) constructing shared knowledge, (2) negotiation / coordination, and (3) maintaining team function. Each facet has three verbal indicators that provide the basis for expert coding (shown in Table 1).

Rather than comprehensively coding all utterances in the dataset, we adopted a thin slicing approach [33] where a random 90 seconds was coded from the first, second, and third five minutes of each 15-minute block (i.e., 30% of all data was coded). Three expert humans were trained to code the utterances for the presence of each verbal indicator. Coders watched videos of the collaborations (for the full context), alongside the automated transcripts, and counted the number of times each indicator occurred in an utterance (>99% of the individual indicator counts were 0 or 1). Coder agreement on the indicators ranged from .88 to 1.00 (Gwet's AC1 metric [22]) on ten 90-second video samples consisting of 406 utterances. After training and achieving adequate reliability, videos were then randomly assigned to the three coders for independent coding.

Next, we used the coded indicator counts to create binary labels for each facet. If all the indicator counts for a facet were 0, then that facet was coded as a 0 (no evidence for the facet). Otherwise, if at least one of the indicators occurred, it was coded as a 1 (positive example of the facet). In total, there were 31,533 utterances coded (75.6% from Physics Playground blocks) with about half being scored for at least one indicator. We defined a fourth binary label, "No Facet Coded", for utterances that were coded as 0 for all three facets. The average team-level base rates of each label for both tasks are shown in Table 1. Note that the percentages sum to over 100%, as it is possible for an utterance to be a positive instance of more than one facet (2.9% and 2.6% of utterances were positive for multiple facets in the Physics and Minecraft data, respectively). Two-tailed paired-samples t-tests on the 87 teams with data in both tasks indicated statistically significant differences (between tasks) in the base rates of all facets except negotiation / coordination (see Table 1).

6 NATURAL LANGUAGE PROCESSING AND SUPERVISED MACHINE LEARNING TECHNIQUES

We tested three different supervised classifiers to predict the human-coded facet for each utterance, using the automatically generated transcripts and human annotations. The three methods are illustrated in Figure 2

6.1 Deep Transfer Learning with BERT

We considered a deep transfer learning approach using the popular Bidirectional Encoder Representations from Transformers (BERT) model [11]. This method involved beginning with a pre-trained BERT model (pre-trained on large amounts of unlabeled text data, specifically the BooksCorpus and English Wikipedia totaling 3.3 Billion words [11]), then fine-tuning the model on our task of predicting CPS facets from utterance transcripts. BERT processes the transcribed utterances using WordPiece tokenization [42], which entails splitting each utterance into a sequence of words, or parts of words. Each unique word or word piece is then converted to an integer (aka., a token) according to BERT's vocabulary. Next, special tokens ([CLS] and [SEP]) are appended to the beginning and end of this sequence, and the sequence is provided as input

Table 1: Verbal indicators for each CPS facet, along with example utterances (IBM Watson transcript) from both tasks. Task-specific language is bolded. Note: PP% = Physics Playground base rate, MC% = Minecraft base rate.

CPS Facet [PP%,MC%]	Verbal Indicators	Physics Example	Minecraft Example
Constructing Shared Knowledge [26.0%, 34.5%] p < .001	Talks about the challenge situation	we only got like two minutes left so we might be able to get all these **levels**	yes so somehow we need to get to the **water**
	Proposes specific solutions	I would also build a **ramp** in case you don't time it correctly	and then you have to **move forward** to **strip block** again
	Confirms understanding	yeah a little bit **longer** and make the **weight** a little a **bigger** yeah	it's **three** yeah yeah it is **three**
Negotiation/ Coordination [14.7%, 15.1%] p = .588	Provides reasons to support a solution	I don't think it's gonna move because it's going to get stuck	because it'll go into the **water** I think otherwise
	Responds to others' questions/ideas	yeah sure	okay
	Discusses the results	it doesn't really go down enough to **spring** it up	like why is he just falling in the **water**
Maintaining Team Function [10.4%, 8.0%] p < .001	Asks others for suggestions	so what would we have to do here	okay so where do I have to put **repeat blocks**
	Compliments or encourages others	good idea	genius
	Provides instructions	restart the **level**	okay can you **run** that real quick
No Facet Coded [51.9%, 44.9%] p < .001		this one looks promising this looks like our skill set	yeah yeah I'm just trying to figure out who is right here
		I don't know	yeah I'm confused of those

to BERT. BERT then maps each input token to a 768-dimensional embedding, which serves as a semantic representation of the token. The embedding of the special [CLS] token captures a representation of the entire sequence of input tokens, and is intended for use in text classification tasks [11]. Finally, to make a classification, we used the embedding of the [CLS] token as input to a single fully connected layer, which outputs predicted probabilities for the given CPS facet. We used the transformers [59] library's implementation of the BertModel with the "bert-base-uncased" pre-trained weights, and used the BertTokenizer to process our utterances. We fine-tuned the models for four epochs using a batch size of 32, based on recommendations from [11].

6.2 Open Vocabulary Approach: Random Forest with N-Gram Features

Next we trained Random Forest classifiers using n-grams features, where counts of words and phrases are used as features for the classifier. We tested unigrams (words) and bigrams (two-word phrases), but not trigrams and beyond due to the sparsity of unique multi-word phrases. To generate these features, we first tokenized each utterance using the nltk [4] tokenizer. Then, counts of n-grams were generated and used as features for the Random Forest classifier, using the scikit-learn [35] library's implementation. We explored various hyperparameters for this model, namely: n-gram range (unigrams, bigrams, or both), whether to remove stop words (i.e., commonly used words such as "a" or "the") [60], the number of estimators (i.e., number of decision trees in the forest), and the maximum depth of each tree.

6.3 Dictionary-based Approach: Random Forest with LIWC Features

Finally we trained Random Forest classifiers using features derived from the Linguistic Inquiry Word Count (LIWC) [57]. To do so, we used our utterance transcripts to generate counts in 93 predefined LIWC categories, which were provided as features to the Random Forest classifier. Similarly to the n-gram models, we explored the number of estimators and the maximum tree depth as hyperparameters.

7 MACHINE LEARNING EXPERIMENTS

We conducted a series of experiments to evaluate model accuracy and generalizability across tasks.

7.1 Data Sampling

A significant obstacle to creating task-generalizable models is domain shift. Domain shift can take two forms: (1) Prior (or base rate) shift, where the distribution of labels differs from one domain to another, and (2) covariate (or feature) shift, where the distribution of features (i.e., the language used) differs between domains [27]. Although there is evidence of prior shift between our two tasks for each label except negotiation / coordination (see Table 1), in this study we focus on feature shift in an effort to understand how the language elicited in one task generalizes to the other. Accordingly, we randomly down-sampled (without replacement) both the Physics and Minecraft datasets until the base rate of each CPS facet was 25%, thus ensuring no shift in the prior probabilities of our

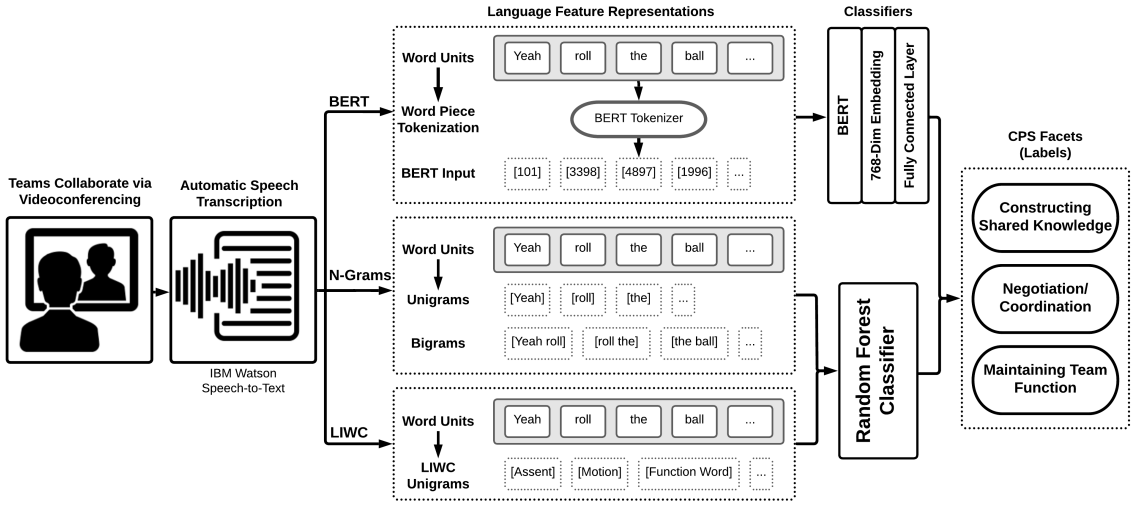


Figure 2: Diagram illustrating our three methods of language feature representation and utterance classification.

labels between the two tasks. This resulted in sampled datasets containing 2,544 utterances, with 636 positive instances of each facet. The remaining utterances (between 30-33%, depending on the number of sampled instances that were positive for multiple facets) were sampled from the "no facet" instances. We performed ten iterations of this random down-sampling to create ten sampled datasets of both Physics and Minecraft data for use in our generalizability experiments. We note that the purpose of sampling is to isolate the effect of language shift between tasks on the generalizability of our three different approaches. It is not to obtain an objective measure of model accuracy, for which we also performed experiments without sampling (Section 7.5). Similarly, we sampled each facet to 25% in order to compare differences in generalizability of the facets without the confound of differing base rates.

7.2 Team-level Cross Validation

We utilized team-level, 10-fold cross validation (Figure 3) to assess model accuracy (within-task evaluation). Team-level cross validation is important for evaluating generalizability across teams (within the same task) because it ensures a model is never trained and tested on utterances from the same team. The process involves training a model on data from 90% of teams, then evaluating the model's predictive accuracy on a test set containing data from the 10% of withheld teams. This is then repeated ten times, with every team appearing in the test set exactly once. Accuracy metrics are computed by aggregating test set predictions from each fold. Thus, this procedure enables us to evaluate how well a model generalizes to data from unseen teams (within the same task context).

We also used a team-level, 10-fold cross validation procedure to evaluate model generalizability (across-task evaluation). However, here we used data from different task contexts in the training and test sets. For example, to evaluate generalizability from Physics to Minecraft, we trained a model on Physics data from 90% of teams, then evaluated the model's predictive accuracy on a test set containing Minecraft data from the 10% of teams withheld during training (repeated ten times as above). This procedure allows us to assess generalizability from one task context to another, while also ensuring team-level generalizability. Importantly, the testing data was the same for both analyses.

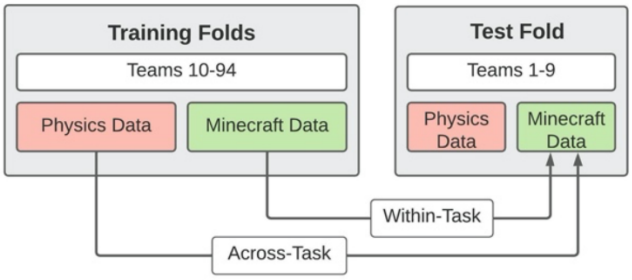
For both the n-gram and LIWC models, we tuned hyperparameters within each training fold using nested five-fold cross validation. To do so, the training set was split into five validation folds (again at the team-level), and for each fold a model was fit using every combination of hyperparameters. The hyperparameters that resulted in the highest average AUROC (across the five validation folds) were preserved, then the model was trained on the full training set for that fold. We did not perform nested hyperparameter tuning with BERT.

7.3 Generalizability Experiments

We used the sampled Physics and Minecraft datasets described above (7.1) to train within-task models for ten iterations, using different randomized team-level cross validation folds for each iteration. We trained our three models (BERT, n-grams, LIWC) using identical cross validation folds in each iteration, to ensure that differences in performance were not a result of the folds used. We trained the models to predict each of the three CPS facets separately (i.e., we used single label rather than multi-label learning), and we also included "No Facet" as a fourth label. After training our within-task models on the sampled datasets, we evaluated their across-task generalizability as noted above. Thus, we had ten iterations of results from both within-task evaluations (i.e., Train Physics – Test Physics and Train Minecraft – Test Minecraft) and both across-task evaluations (i.e., Train Physics – Test Minecraft and Train Minecraft – Test Physics).

7.4 Task Prediction Experiment

To further investigate the generalizable properties of our models, we conducted an experiment wherein we combined the sampled Physics and Minecraft datasets and trained the three models to



$$(1) \qquad \mathrm{TR} = \frac{AUROC_{across\ task} - .50}{AUROC_{within\ task} - .50}$$

$$(2) \qquad \mathrm{TR} = \frac{AUROC_{Train\ PP,\ Test\ MC} - .50}{AUROC_{Train\ MC,\ Test\ MC} - .50}$$

Figure 3: Diagram illustrating within- and across- task evaluation (left), and the computation of Transfer Ratio (TR) (right). (1) shows the general formulation of TR, and (2) shows TR quantifying generalizability from Physics to Minecraft data (elaborated in 7.6).

predict the task context itself (rather than the CPS facets). This experiment was informed by domain adaptation theory, which suggests that cross-domain generalization can be achieved through feature representations from which the domain of an input example cannot be identified [3]. In other words, if a model cannot accurately differentiate between instances from the two tasks, it is likely to generalize across the tasks. Accordingly, we combined the sampled Physics and Minecraft datasets within each of the ten iterations, resulting in ten datasets of 5,088 utterances. The combined datasets contained an equal number of Minecraft and Physics utterances (2,544 each), and each CPS facet had a base rate of 25%. Then, for each combined dataset, we trained the three models to predict the task context (i.e., predict if the utterance came from a Physics block or a Minecraft block), using 10-fold team-level cross validation to evaluate the accuracy of predicting the task context.

7.5 Evaluation on Full Datasets

Because the down-sampling described above (7.1) significantly reduced the amount of data used in our experiments (2,544 utterances represents 11% of the Physics data and 33% of the Minecraft data), we also trained within-task models (as described in 7.3) on the full Physics and Minecraft datasets (23,825 and 7,708 utterances, respectively). We conducted five iterations of this experiment in order to determine how model results from the sampled datasets compare with model results on the full datasets.

7.6 Metrics

We used the area under the receiver operating characteristic curve (AUROC) [5] to assess model accuracy. The models output a predicted probability between 0 and 1 that each utterance is a positive instance of the given facet. The AUROC considers the true positive and false positive tradeoff across classification thresholds rather than selecting a single probability threshold for binary classification. An AUROC of 1 represents perfect classification, and an AUROC of .5 represents chance performance. In order to quantify model generalizability across tasks, we use the Transfer Ratio (TR) [19], which quantifies the relative decrease in performance of a model trained and tested on data from different tasks (i.e., across-task evaluation) versus the same task (i.e., within-task evaluation). Thus, a TR of 1 indicates perfect generalizability, (no decrease in performance due to across-task evaluation). The general formulation of the TR is presented above in Figure 3 (Equation 1). Note that .50 is deducted from both numerator and denominator to quantify the difference in performance over chance. We specifically compute the TR at the level of the task used for model testing. For example, the TR quantifying generalizability from Physics to Minecraft is computed as shown above in Figure 3 (Equation 2).

8 RESULTS

8.1 Accuracy and Generalizability

Results from our generalizability experiments (Section 7.3) are shown in Table 2 and Table 3. For each iteration, we computed the metrics separately for each facet (as described in Section 7.6) and then computed the row and column means by averaging across facets or by averaging across models. Finally, we averaged values over the ten iterations. In addition, Figure 4A shows the full distributions of AUROC values for both within-task (green) and across-task (orange) evaluation. The corresponding distributions of TRs (grey) are plotted alongside (Figure 4B). Each distribution contains twenty values (ten iterations of both Physics and Minecraft data).

We found that when averaging across facets (row means), the BERT and LIWC models performed similarly in terms of accuracy and generalizability (equivalent within-task AUROCs [.79] and TRs [.93]), and outperformed the n-gram models (AUROC of .77 and TR of .86). Both accuracy (AUROC of .75) and generalizability (TR of .86) were lower for maintaining team function compared to the other two facets (AUROCs > .79; TRs > .90). Interestingly, the best generalizability (TR of .95) was achieved on the "no facet" instances, suggesting that the language indicative of "no facet" instances is relatively invariant across task contexts. These results indicate that given the correct choice of model, task-generalizability of CPS facets is feasible, although some degradation in performance still occurs.

8.2 Task Prediction

Results from our task prediction experiment (Section 7.4) are presented in Table 4 (left). These results are interesting in light of our previous findings on model generalizability (see 8.1). The LIWC model, which generalized well across tasks (TR = .93), was less able to distinguish Physics vs. Minecraft utterances (AUROC of .69), while the n-gram model, which did not generalize as well (TR = .86),

Table 2: AUROC values for each CPS facet (plus no facet). The first value in each triplet represents evaluation on Physics data, the second represents evaluation on Minecraft data, and the last (bolded) value is the average over the two tasks.

	Constructing Shared Knowledge		Negotiation/ Coordination		Maintaining Team Function		No Facet		Mean	
Model	Within	Across	Within	Across	Within	Across	Within	Across	Within	Across
BERT	.80 .80 **.80**	.78 .77 **.77**	.78 .80 **.79**	.77 .78 **.77**	.76 .77 **.76**	.72 .74 **.73**	.79 .80 **.79**	.78 .79 **.78**	.78 .79 **.79**	.76 .77 **.77**
NGram	.80 .81 **.80**	.77 .74 **.75**	.79 .79 **.79**	.74 .76 **.75**	.73 .74 **.74**	.69 .70 **.69**	.76 .77 **.77**	.74 .74 **.74**	.77 .78 **.77**	.73 .74 **.74**
LIWC	.81 .80 **.80**	.79 .78 **.78**	.79 .80 **.80**	.76 .78 **.77**	.74 .78 **.76**	.72 .74 **.73**	.79 .79 **.79**	.78 .78 **.78**	.78 .79 **.79**	.76 .77 **.77**
Mean	.80 .80 **.80**	.78 .76 **.77**	.79 .80 **.79**	.76 .77 **.77**	.74 .76 **.75**	.71 .73 **.72**	.78 .79 **.78**	.77 .77 **.77**	.78 .79 **.78**	.75 .76 **.76**

Table 3: TR values for each CPS facet (plus no facet). See Table 2 caption regarding triplet values.

Model	Constructing Shared Knowledge	Negotiation/ Coordination	Maintaining Team Function	No Facet	Mean
BERT	.91 .91 **.91**	.97 .93 **.95**	.88 .88 **.88**	.98 .98 **.98**	.93 .92 **.93**
NGram	.89 .80 **.84**	.84 .88 **.86**	.79 .84 **.82**	.93 .87 **.90**	.86 .85 **.86**
LIWC	.96 .91 **.93**	.91 .92 **.91**	.91 .87 **.89**	.98 .97 **.97**	.94 .92 **.93**
Mean	.92 .87 **.90**	.91 .91 **.91**	.86 .87 **.86**	.96 .94 **.95**	.91 .90 **.90**

Table 4: AUROC values for the task prediction experiment, partitioned by facet (left) and for experiment comparing sampled. vs. full models (right). Values reported are the mean over all iterations of each experiment. Note: Const. = constructing shared knowledge, Neg. = negotiation / coordination, Maintain. = maintaining team function.

	Task prediction experiment					**Sampled vs. Full Datasets**			
						Minecraft		Physics	
Model	Const.	Neg.	Maintain.	No Facet	All	Sampled (n=2,544)	Full (n=7,708)	Sampled (n=2,544)	Full (n=23,825)
BERT	.76	.77	.77	.77	.77	.79	.82	.78	.83
Ngram	.75	.76	.76	.76	.76	.78	.79	.77	.79
LIWC	.70	.69	.70	.69	.69	.79	.81	.78	.81

achieved a higher AUROC of .76. This finding is consistent with domain adaptation theory, which suggests that feature representations which are less predictive of the domain of origin are more likely to generalize across domains [3]. However, the BERT models, which showed similar across-task generalizability to LIWC (TR = .93), achieved an AUROC (.77) on par with the n-gram models. We also did not find differences in task prediction accuracy between the CPS facets, suggesting similar task-specific language across facets.

8.3 Full vs. Sampled Datasets

Results for comparing within-task accuracy on the sampled and full datasets (Section 7.5) are shown in Table 4 (right). The value presented is the mean AUROC over all four labels. As expected, accuracy of all three models increased on the full datasets. The BERT model saw the largest gain in performance (+.03, +.05 AUROC on Minecraft and Physics datasets respectively), followed by the LIWC model (+.02, +.03 AUROC), then the n-gram model (+.01, +.02 AUROC). These findings indicate that deep learning models such as BERT may have an advantage (in terms of accuracy) over traditional classifiers when more training data is available [26]. That said, the magnitude of improvements were not proportional to the amount of additional data available in the full datasets (approximately 3x more data for Minecraft, 9x for Physics). Overall, this provides evidence that the conclusions obtained from the experiments with sampled datasets are generalizable to the full dataset.

9 DISCUSSION

We investigated differences in the accuracy and generalizability of three different NLP classifiers trained to predict CPS facets from automatically generated transcripts. In this section, we discuss our main findings, applications of this work, the limitations of this study and future directions for research.

9.1 Main Findings

Our findings indicate that language-based models of CPS can generalize between two distinct task contexts with only a small degradation in performance (best models achieved TR of .93). We hypothesized the following pattern for across-task generalizability *BERT > LIWC > N-Grams*, but found instead: *[BERT = LIWC] > N-Grams*. We observed the same pattern for within-task accuracy although we hypothesized *BERT > N-Grams > LIWC* . This finding is noteworthy

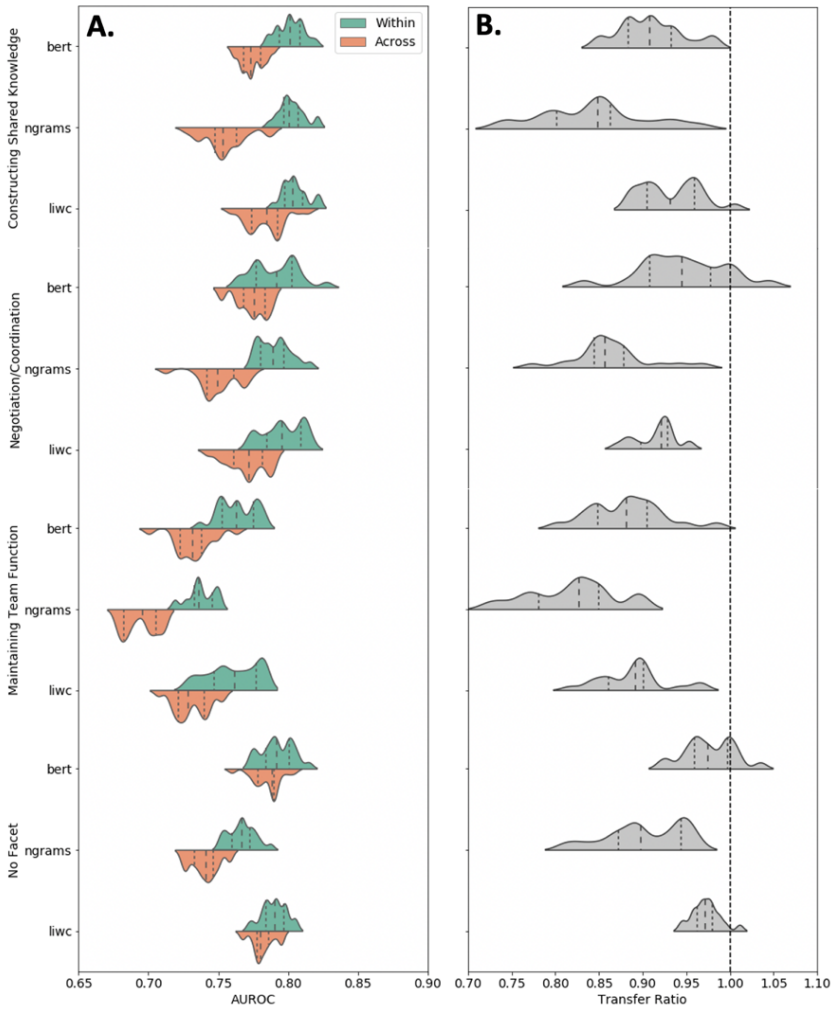


Figure 4: (A) Distributions of AUROC values for each model and facet. The top distribution shows within-task evaluation, and the bottom distribution shows across-task evaluation. (B) Corresponding distributions of Transfer Ratio values, as described in section 7.6. The vertical dashed line represents a TR of 1 (perfect generalizability across tasks).

because many previous studies on language-based collaboration analytics (e.g., [18, 40, 51]) have used n-grams, yet we found this approach to yield both the worst accuracy and generalizability. It is also notable that the LIWC model performed on par with the state-of-the-art BERT model for both accuracy and generalizability when trained on the sampled datasets.

Our task prediction experiment demonstrated that the BERT and n-gram models were better able to distinguish between utterances from the two tasks than the LIWC model. The BERT and n-gram models consider the actual words contained in an utterance, and therefore capture more task-specific information. For instance, use of the words “draw”, “launch”, or “ball” indicate the Physics task, while words like “turn” or “brick” indicate the Minecraft task. However, this relationship is obfuscated by the LIWC feature representation, which converts words into counts of predefined word categories. For example, the Physics-specific word “launch” is replaced with the more general word categories “motion”, “relativity”, “causation”, “cognitive process”, and the Minecraft-specific word “turn” is replaced with the categories “motion”, “relativity”, “verb”, “present focus”. This additional layer of abstraction explains the LIWC model’s inferior performance in the task prediction experiment and may contribute to its ability to generalize across task contexts.

Interestingly, although the BERT model achieved the highest task prediction accuracy, it also generalized as well as the LIWC model. We hypothesize that BERT’s ability to learn sophisticated semantic representations of language (as evidenced by its state-of-the-art performance in various benchmark NLP tasks [11]) enables it to identify language indicative of CPS facets in a more task-independent manner. BERT also outperformed the other models (in terms of within-task accuracy) when trained on the full datasets.

Taken together, these findings suggest that large pre-trained language models such as BERT may be more suitable when large quantities of data are available, while a dictionary-based approach such as LIWC may be more appropriate with smaller datasets.

9.2 Applications, Limitations, and Future Work

An important application of this work includes utilizing language-based collaboration analytics (CA) models in authentic educational settings to provide automated formative assessment of CPS skills and interventions to improve said skills. For instance, automated reports could be displayed to a teacher monitoring many groups of students engaged in CPS (e.g., via a teacher dashboard), informing the teacher of the extent to which each group is engaging in different aspects of CPS (e.g., constructing shared knowledge). Such a system could help the teacher identify groups that need their support and know how to best allocate their limited presence. Likewise, the system could help a teacher identify individual students' strengths and weaknesses to set appropriate goals for improvement. For such applications (in school environments), collaborative activities are not restricted to a single task context (as most lab studies are). For example, a teacher may regularly update a group learning or problem solving activity, changing the task context or problem solving affordances to meet their pedagogical objectives or the needs of individual classes. It is not realistic to expect task-specific data (transcripts) to be available (i.e., to train task-specific models or perform domain adaptation techniques) for every context in which CA models will be deployed. Rather, a more practical approach is to pursue representations of collaborative language which generalize across different tasks as we have done here.

In addition to teacher-facing analytics, this approach could be used to provide learner-facing feedback aimed at developing CPS skills. For instance, CA models could display insights to individual team members, illustrating how well they contributed to their team and demonstrated different CPS skills. This kind of personalized feedback could enhance learners' self-awareness, reflection, and evaluation of their strengths and weaknesses, and help them track their improvement in different skills across a series of collaborations.

Like all studies, ours has limitations. To begin, we only evaluated model generalizability between two CPS tasks, both of which were conducted in a remote, computerized setting (via videoconferencing with screen share). Thus, it's possible that our findings will not generalize to other contexts, such as co-located face-to-face CPS, where collaboration occurs in a shared physical space and involves interaction with tangible (rather than digital) artifacts. Future work will examine generalizability between additional (more dissimilar) task contexts. Similarly, in this work we only investigated generalizability between different tasks, and did not explore generalizability between different blocks of the same task (i.e., between the first block of Physics Playground and the third block of Physics Playground, which may have relevant differences due to familiarity with team members, etc.).

Another limitation is that we did not investigate the effect that differing facet base rates had on generalizability between the two tasks. Rather, we sampled each facet to a constant 25% to isolate the effects of feature (language) shift between the tasks and to compare several feature representations. Future work will need to address the challenge of base rate shift between tasks and examine its effect on model generalizability. Further, ideally the order of the two tasks would have been counterbalanced across teams, but this was not feasible since we used an existing dataset. However, we do not think this influenced the results since across-task generalizability was similar for both cases (train Physics, test Minecraft and vice versa; see Table 3). Finally, we only investigated one application – the modeling of CPS facets. Future work should examine whether language-based CA models with different applications (e.g., predicting task performance, identifying phases of collaboration) will also generalize across tasks.

10 CONCLUSION

We demonstrated the feasibility of training speech-based models of CPS facets that generalize across different task contexts. Our findings indicate that the choice of feature representation used to model collaborative language is important to obtaining task-generalizable models. This work contributes to the broader goal of deploying collaboration analytics models in authentic educational environments.

ACKNOWLEDGMENTS

This research was supported by the NSF National AI Institute for Student-AI Teaming (iSAT) (DRL 2019805), NSF DUE 1745442/1660877, and Institute of Educational Sciences (IES R305A170432). The opinions expressed are those of the authors and do not represent views of the funding agencies.

REFERENCES

[1] Andrews-Todd, J. and Forsyth, C.M. 2020. Exploring social and cognitive dimensions of collaborative problem solving in an open online simulation-based task. *Computers in Human Behavior*. 104, (2020). DOI:https://doi.org/10.1016/j.chb.2018.10.025.

[2] Baker, R.S.J.D. *et al.* 2008. Developing a generalizable detector of when students game the system. *User Modeling and User-Adapted Interaction*. 18, 3 (2008). DOI:https://doi.org/10.1007/s11257-007-9045-6.

[3] Ben-David, S. *et al.* 2010. A theory of learning from different domains. *Machine Learning*. 79, 1–2 (2010).

[4] Bird, S. *et al.* 2016. NLTK: The natural language toolkit NLTK: The Natural Language Toolkit. *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1*. March (2016).

[5] Bradley, A.P. 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*. 30, 7 (1997).

[6] Chejara, P. *et al.* 2021. Efar-mmla: An evaluation framework to assess and report generalizability of machine learning models in mmla. *Sensors*. 21, 8 (2021). DOI:https://doi.org/10.3390/s21082863.

[7] Chopade, P. *et al.* 2019. CPSX: Using AI-Machine Learning for Mapping Human-Human Interaction and Measurement of CPS Teamwork Skills. *2019 IEEE International Symposium on Technologies for Homeland Security, HST 2019* (2019).

[8] Code Studio. Retrieved August 9, 2021 from https://studio.code.org/s/mc/stage/1/puzzle/1: .

[9] Crossley, S. *et al.* 2017. Predicting math performance using natural language processing tools. *ACM International Conference Proceeding Series* (2017).

[10] von Davier, A.A. *et al.* 2017. Interdisciplinary research agenda in support of assessment of collaborative problem solving: lessons learned from developing a Collaborative Science Assessment Prototype. *Computers in Human Behavior*. 76, (2017).

[11] Devlin, J. *et al.* 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* (2019).

[12] Dowell, N.M.M. *et al.* 2020. Exploring the relationship between emergent sociocognitive roles, collaborative problem-solving skills, and outcomes: A group

communication analysis. *Journal of Learning Analytics*. 7, 1 (2020). DOI:https://doi.org/10.18608/jla.2020.71.4.
[13] Dowell, N.M.M. *et al.* 2019. Group communication analysis: A computational linguistics approach for detecting sociocognitive roles in multiparty interactions. *Behavior Research Methods*. 51, 3 (2019). DOI:https://doi.org/10.3758/s13428-018-1102-z.
[14] E. Bixler, R. and K. D'Mello, S. 2021. Crossed Eyes: Domain Adaptation for Gaze-Based Mind Wandering Models. *Eye Tracking Research and Applications Symposium (ETRA)* (2021).
[15] Emara, M. *et al.* 2021. Examining Student Regulation of Collaborative, Computational, Problem-Solving Processes in Open-Ended Learning Environments. *Journal of Learning Analytics*. 8, 1 (2021). DOI:https://doi.org/10.18608/jla.2021.7230.
[16] Fiore, S.M. *et al.* 2018. Collaborative problem-solving education for the twenty-first-century workforce. *Nature Human Behaviour*.
[17] Fiore, S.M. *et al.* 2017. Collaborative Problem Solving: Considerations for the National Assessment of Educational Progress. *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing - CSCW '14*. (2017).
[18] Flor, M. *et al.* 2016. Automated classification of collaborative problem solving interactions in simulated science tasks. *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications* (2016).
[19] Glorot, X. *et al.* 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. *Proceedings of the 28th International Conference on Machine Learning, ICML 2011* (2011).
[20] Graesser, A.C. *et al.* 2018. Advancing the Science of Collaborative Problem Solving. *Psychological Science in the Public Interest*. 19, 2 (2018), 59–92.
[21] Graesser, A.C. *et al.* 2004. Coh-Metrix: Analysis of text on cohesion and language. *Behavior Research Methods, Instruments, and Computers* (2004).
[22] Gwet, K.L. 2010. Handbook of Inter-Rater Reliability: the definitive guide to measuring the extent of agreement among raters.
[23] Hao, J. *et al.* 2017. CPS-Rater: Automated Sequential Annotation for Conversations in Collaborative Problem-Solving Activities. *ETS Research Report Series*. 2017, 1 (2017). DOI:https://doi.org/10.1002/ets2.12184.
[24] Hutt, S. *et al.* 2019. Time to scale: Generalizable affect detection for tens of thousands of students across an entire school year. *Conference on Human Factors in Computing Systems - Proceedings* (2019).
[25] IBM Watson: https://www.ibm.com/watson/services/speech-to-text/. Accessed: 2021-03-02.
[26] Jensen, E. *et al.* 2021. A deep transfer learning approach to modeling teacher discourse in the classroom. *ACM International Conference Proceeding Series* (2021).
[27] Kouw, W.M. and Loog, M. 2021. A Review of Domain Adaptation without Target Labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
[28] Lämsä, J. *et al.* 2021. Deep Networks for Collaboration Analytics: Promoting Automatic Analysis of Face-to-Face Interaction in the Context of Inquiry-Based Learning. *Journal of Learning Analytics*. 8, 1 (2021). DOI:https://doi.org/10.18608/jla.2021.7118.
[29] Lubold, N. and Pon-Barry, H. 2014. Acoustic-prosodic entrainment and rapport in collaborative learning dialogues. *MLA 2014 - Proceedings of the 2014 ACM Multimodal Learning Analytics Workshop and Grand Challenge, Co-located with ICMI 2014* (2014).
[30] Martinez-Maldonado, R. *et al.* 2019. Collocated Collaboration Analytics: Principles and Dilemmas for Mining Multimodal Interaction Data. *Human-Computer Interaction*. 34, 1 (2019). DOI:https://doi.org/10.1080/07370024.2017.1338956.
[31] OECD 2015. Pisa 2015 Collaborative Problem Solving Framework. (2015).
[32] OECD 2017. PISA 2015 Results (Volume V): Collaborative Problem Solving.
[33] Olsen, J.K. and Finkelstein, S. 2017. Through the (Thin-slice) looking glass: An initial look at rapport and co-construction within peer collaboration. *Computer-Supported Collaborative Learning Conference, CSCL* (2017).
[34] Patikorn, T. *et al.* 2019. Generalizability of methods for imputing mathematical skills needed to solve problems from texts. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2019).
[35] Pedregosa, F. *et al.* 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*. 12, (2011).
[36] Pugh, S.L. *et al.* 2021. Say What? Automatic Modeling of Collaborative Problem Solving Skills from Student Speech in the Wild. *Proceedings of The 14th International Conference on Educational Data Mining (EDM21)* (2021), 55–67.
[37] Ramponi, A. and Plank, B. 2021. Neural Unsupervised Domain Adaptation in NLP—A Survey. (2021).
[38] Reilly, J.M. and Schneider, B. 2019. Predicting the quality of collaborative problem solving through linguistic analysis of discourse. *EDM 2019 - Proceedings of the 12th International Conference on Educational Data Mining* (2019).
[39] Roschelle, J. and Teasley, S.D. 1995. The Construction of Shared Knowledge in Collaborative Problem Solving. *Computer Supported Collaborative Learning*.
[40] Rosé, C. *et al.* 2008. Analyzing collaborative learning processes automatically: Exploiting the advances of computational linguistics in computer-supported collaborative learning. *International Journal of Computer-Supported Collaborative Learning*. 3, 3 (2008).
[41] Schneider, B. *et al.* 2021. Collaboration Analytics — Current State and Potential Futures. *Journal of Learning Analytics*. 8, 1 (2021).
[42] Schuster, M. and Nakajima, K. 2012. Japanese and Korean voice search. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings* (2012).
[43] Schwartz, H.A. *et al.* 2013. Personality, Gender, and Age in the Language of Social Media: The Open-Vocabulary Approach. *PLoS ONE*. 8, 9 (2013).
[44] Sharma, K. *et al.* 2020. Assessing Cognitive Performance Using Physiological and Facial Features. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*. 4, 3 (2020). DOI:https://doi.org/10.1145/3411811.
[45] Shute, V.J. *et al.* 2013. Assessment and learning of qualitative physics in Newton's playground. *Journal of Educational Research*. 106, 6 (2013).
[46] Shute, V.J. 2008. Focus on formative feedback. *Review of Educational Research*. 78, 1 (2008). DOI:https://doi.org/10.3102/0034654307313795.
[47] Sinclair, A.J. and Schneider, B. 2021. Linguistic and Gestural Coordination: Do Learners Converge in Collaborative Dialogue? *Proceedings of the 14th International Conference on Educational Data Mining (EDM 2021)* (2021).
[48] Skalicky, S. *et al.* 2017. Identifying Creativity During Problem Solving Using Linguistic Features. *Creativity Research Journal*. 29, 4 (2017).
[49] Stewart, A. *et al.* 2017. Generalizability of Face-Based Mind Wandering Detection across Task Contexts. *EDM*. (2017).
[50] Stewart, A.E.B. *et al.* 2020. Beyond Team Makeup: Diversity in Teams Predicts Valued Outcomes in Computer-Mediated Collaborations. *Conference on Human Factors in Computing Systems - Proceedings* (2020).
[51] Stewart, A.E.B. *et al.* 2019. I say, you say, we say: Using spoken language to model socio-cognitive processes during computer-supported collaborative problem solving. *Proceedings of the ACM on Human-Computer Interaction*. 3, CSCW (2019). DOI:https://doi.org/10.1145/3359296.
[52] Stewart, A.E.B. *et al.* 2021. Multimodal modeling of collaborative problem-solving facets in triads. *User Modeling and User-Adapted Interaction*. (2021).
[53] Stewart, A.E.B. *et al.* 2018. Multimodal modeling of coordination and coregulation patterns in speech rate during triadic collaborative problem solving. *ICMI 2018 - Proceedings of the 2018 International Conference on Multimodal Interaction* (2018).
[54] Subburaj, S.K. *et al.* 2020. Multimodal, Multiparty Modeling of Collaborative Problem Solving Performance. *ICMI 2020 - Proceedings of the 2020 International Conference on Multimodal Interaction* (2020).
[55] Sullivan, F.R. and Keith, P.K. 2019. Exploring the potential of natural language processing to support microgenetic analysis of collaborative learning discussions. *British Journal of Educational Technology*. 50, 6 (2019). DOI:https://doi.org/10.1111/bjet.12875.
[56] Sun, C. *et al.* 2020. Towards a generalized competency model of collaborative problem solving. *Computers and Education*. 143, (2020).
[57] Tausczik, Y.R. and Pennebaker, J.W. 2010. The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology*.
[58] Vrzakova, H. *et al.* 2020. Focused or stuck together: Multimodal patterns reveal triads' performance in collaborative problem solving. *ACM International Conference Proceeding Series* (2020).
[59] Wolf, T. *et al.* 2019. Transformers: State-of-the-art natural language processing. *arXiv*.
[60] Zaman, A.N.K. *et al.* 2011. Evaluation of stop word lists in text retrieval using latent semantic Indexing. *2011 6th International Conference on Digital Information Management, ICDIM 2011* (2011).