Informing Expert Feature Engineering through Automated Approaches: Implications for Coding Qualitative Classroom Video Data

Paul Hur
University of Illinois Urbana–Champaign
USA
khur4@illinois.edu

Nessrine Machaka
University of Illinois Urbana–Champaign
USA
machaka2@illinois.edu

Christina Krist
University of Illinois Urbana–Champaign
USA
ckrist@illinois.edu

Nigel Bosch
University of Illinois Urbana–Champaign
USA
pnb@illinois.edu

ABSTRACT

While classroom video data are detailed sources for mining student learning insights, their complex and unstructured nature makes them less than straightforward for researchers to analyze. In this paper, we compared the differences between the processes of expert-informed manual feature engineering and automated feature engineering using positional data for predicting student group interaction in four middle school and high school mathematics classroom videos. Our results highlighted notable differences, including improved model accuracy for the combined (manual features + automated features) models compared to the only-manual-features models (mean AUC = .778 vs. .706) at the cost of feature interpretability, increased number of features for automated feature engineering (1523 vs. 178), and engineering approach (domain-agnostic in automated vs. domain-knowledge-informed in manual). We carried out feature importance analyses and discuss the implications of the results for potentially augmenting human perspectives about qualitatively coding classroom video data by confirming and expanding views on which body areas and characteristics may be relevant to the target interaction behavior. Lastly, we discuss our study's limitations and future work.

CCS CONCEPTS

• **Applied computing** → **Education**.

KEYWORDS

classroom video data, expert-informed feature engineering, student positional data

ACM Reference Format:
Paul Hur, Nessrine Machaka, Christina Krist, and Nigel Bosch. 2023. Informing Expert Feature Engineering through Automated Approaches: Implications for Coding Qualitative Classroom Video Data. In *LAK23: 13th International Learning Analytics and Knowledge Conference (LAK 2023), March 13–17, 2023, Arlington, TX, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3576050.3576090

LAK 2023, March 13–17, 2023, Arlington, TX, USA

ACM ISBN 978-1-4503-9865-7/23/03…$15.00
https://doi.org/10.1145/3576050.3576090

1 INTRODUCTION

Feature engineering is a crucial step in machine learning which involves brainstorming and transforming raw data to create relevant predictor variables for the target variable. It is often a manual process in the case of expert-informed feature engineering, and benefits from subjective human perspectives and intuition from domain knowledge in order to attain good data representation [13]. Extensive domain expertise alone, however, may not directly translate to effective feature engineering. The most relevant qualitative characteristics related to the target variable may either be too difficult to quantify, calculate, or operationalize, or the data needed for the feature may not exist. Automated feature engineering methods could perhaps be used to aid this gap. Tools such as FeatureTools [22], TSFRESH [9], or AutoFeat [17] do not utilize domain knowledge to create features, but instead consider the nature of data (e.g., column data type, recognizing value patterns such as time or dates, etc.) and apply appropriate transformations based on hierarchical relationships to rapidly generate a large number of features. By examining the similarities and differences between models using manually created features and automated features, there is potential for human perspectives to be guided on the characteristics which are overlooked, yet meaningful to the target variable.

Both quantitative and qualitative education researchers have closely examined classroom video data. Within learning analytics, researchers have used video data to manually label student behaviors for building predictive models for characteristics related to successful learning [5, 19, 30]. In qualitative education research, the complex, unstructured nature of video data allows for deep qualitative analyses of students' behaviors and perspectives in ways which may not be possible with quantitative methods [3]. In order to organize the complexity of video data, researchers commonly rely on qualitative coding to categorize information, find patterns, and extract meaning. Classroom videos have been used to inform pedagogical theories through analyzing teacher–student interactions and dialogue [2], study students' spatial reasoning and

sensemaking [38], and support teacher professional development through self-reflective practice [10, 29], to name a few examples.

Despite the prevalence of qualitative labeling and coding of video data in education research, it is often a laborious and time-consuming process. The process entails determining the appropriate target behaviors based on the available data, behaviors to be defined, coders to be trained in order to align subjective perspectives, and obtain a reasonable degree of inter-rater agreement [16]. Furthermore, due to its manual nature, the amount of effort scales linearly with the amount of data. Recent advancements in machine learning methods and computer vision software have the potential to augment the qualitative coding process. Off-the-shelf automatic video analysis methods such as OpenPose [8] or MediaPipe [27] can be used to collect positional data of students in the classroom video (i.e., x and y coordinates), and when combined with available post-processing methods [1, 18], it is possible to track students' movements. The resulting data could be used to build machine learning models that predict student learning behaviors across class periods, for example.

1.1 Contribution

In this paper, we explore how automated feature engineering methods could inform expert feature engineering processes using individuals' positional data for predicting classroom video behaviors. We compare the differences between a machine learning model built from a expert-informed manual feature set to a model that used a larger feature set combining both manual features and automatically generated features. We aim to explore whether automated feature engineering methods—in particular, FeatureTools—could be used in tandem with manual feature engineering methods to balance human perspectives and computational perspectives. Human perspectives typically inform the creation of new computational methods or tools; here, we are motivated to explore the reverse: how computational methods could inform human perspectives around qualitative classroom video analyses.

Our paper was guided by the following research questions:

- **RQ1** What are the differences between manual vs. automated feature engineering methods for creating features related to predicting student interactions with others?
- **RQ2** In what ways can automated feature engineering methods inform manual feature engineering methods for predicting student interactions with others, and what are the implications for coding qualitative video data?

Next, we briefly outline related work before describing the methods, results, and implications of these research questions.

2 RELATED WORK

In the related work section, we discuss how education researchers have used physical positioning information from video data to study classroom learning, and how domain expertise has been used to create effective features for predictive models.

2.1 Studying the role of physical positioning in learning

The current state of research on examining the role of student and teacher physical positioning shows that it closely relates to the social dynamics around interpersonal ties [11, 36] and power [14, 25], which has implications for pedagogical effectiveness [26, 35] and collaborative learning [7, 32]. There is a breadth of learning analytics work on predicting the emotional, behavioral, and cognitive behaviors of students in the classroom using multi-modal approaches, including positioning data, through position sensors and computer vision tools [1, 12, 28]. In more qualitatively oriented studies, qualitative coding of positional information from video data along with surveys, interviews, field notes, and other learning artifacts have been used to study student engagement and the evolution of student knowledge-building strategies [14, 23]. Our paper closely relates to the aforementioned areas of study by providing potential avenues for augmenting researchers' perspectives around interpreting student positional information.

2.2 Expert-informed feature engineering

Human perspectives informed by domain expertise have been leveraged to create effective features for predicting the target label in a wide variety of classification tasks, such as diagnosing heart disorders [20], phenotyping genome data [37], and detecting emotions during learning [21, 34]. In these studies, experts utilized their understanding of the theoretical underpinnings and empirical observations to create features. Therefore, for the manual feature engineering of student positional data extracted from classroom video data, education researchers' domain expertise and video observation notes could perhaps be leveraged to create effective features related to predicting student group interaction.

3 METHODS

In this section, we discuss target behavior and data selection, OpenPose video processing, qualitative coding, manual and automated feature engineering, as well as model building.

3.1 Video data and target student behavior

The video data used for this research were classroom videos previously collected for a qualitative research project exploring various middle school and high school mathematics teachers' responsive teaching practices. Cameras positioned at the corners of classrooms were used to capture students seated in small groups and the teacher at 1080p (1920×1080 pixels) resolution at 30 frames per second with 120 or 130-degree fields of view. Based on our observations of watching the various classroom videos, we decided to qualitatively code for the presence or absence of student group interaction as it relates closely to collaborative learning [24]. Students are able to develop higher level thinking skills by working together through sharing, discussing ideas, and receiving peer feedback [33]. *Interaction* (*i*) vs. *No interaction* (*n*) would be sufficiently low-level to code for a large number of occurrences, while being high-level to be a relevant behavior for qualitative education video research. Prior to the coding process, we defined code definitions and examples activities which are presented in Table 1.

Table 1: *Interaction* (*i*) vs. *No interaction* (*n*) code definitions.

Code	Definition	Examples
Interaction (*i*)	One or more students in the group are interacting with other students or the teacher	Talking, actively listening, gesturing
No interaction (*n*)	No students in the group are interacting with other students or the teacher	Writing/working individually, looking down (no one speaking toward them)

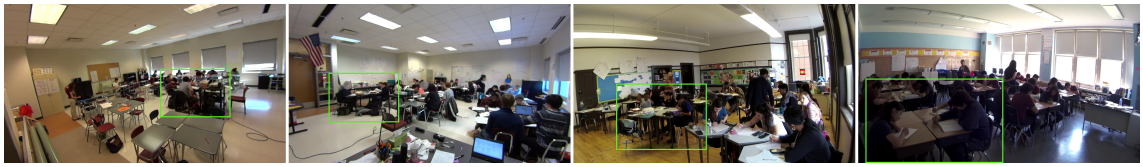

Figure 1: Video frames from each class period from left to right: Class A, Class B, Class C, and Class D. Target student groups are indicated in green.

We selected four different videos of ~90-minute class periods (two high school, two middle school), and selected one target student group to code from each video. We focused on selecting diverse learning environments and target student group location—Class A, Class B, Class C, Class D are shown in Figure 1. Then, from each video, we trimmed one continuous 9-minute segment of individual work (e.g., teacher announces to the class that students will be working to solve a problem independently) and one continuous 9-minute segment of small group work (e.g., teacher announces to the class that students should work and discuss their work in their small groups), for a total of 18 minutes per video. Trials of different coding clip lengths (i.e., 5, 10, 15, and 30 seconds) revealed that 10-second coding clip lengths balanced brief interactions without including multiple interaction codes, while still providing enough context to code longer interactions. Thus, every minute of video had 6 codes (*i* or *n*) for a total of 108 codes per class session.

Two trained coders independently coded one set of videos (two 9-minute segments) in order to establish inter-coder agreement. The coders agreed on 93.5% (101 out of 108, kappa = 0.87) of the labels. After establishing this agreement, the rest of the videos were coded separately. Each coder kept notes on how they determined the codes, and listed characteristics which helped them to determine whether or not interaction was occurring.

3.2 Video processing and student positional data

We used the computer vision tool OpenPose [8] to process and extract positional data from the videos by identifying individuals' body parts as ordered keypoints. For our analysis, we used OpenPose's 25-keypoint body configuration, such that each individual of each frame of video has a maximum of 25 keypoints. The output file formats individuals' keypoint data as x and y coordinate values in terms of the video's resolution (i.e., 1920×1080 in this case). We processed each video through OpenPose, and post-processed the output files to track individuals' movements over time using an open-source OpenPose data tracking method [18] and restricted the tracking calculation to the target student group region in the video (groups indicated in green in Figure 1). This allowed us to obtain relationships in the data by assigning person IDs to individuals by determining Euclidean distances of available respective keypoints and connecting the closest matches between frames to the person ID. The process outputted a CSV file where each row represents a person detected per frame with the following information: person ID, frame number, whether each keypoint was detected, as well as x and y coordinate information for each of the 25 keypoints.

3.3 Manual and automated feature engineering

We brainstormed features using the notes that trained video coders had taken to determine the group interaction coding labels (summarized in Table 2). Coders had primarily noticed that group interactions were typically indicated by increased movement: frequent gesturing, heads moving while lips also moved when speaking and exchanging objects or learning material. On the other hand, a lack of interaction was indicated by reduced movement and less visibility of group members due to leaning down toward their desks and working separately. We aligned codes to the frame-level tracked OpenPose output data, and created 178 primarily movement-focused features, each created at the clip-level. Features were separately calculated for each of the four class periods, as different camera angles meant that the each video's positional data were scaled differently. We calculated such features as: clip-level (10 second) maximums and means of individuals' keypoint x and y coordinate movement magnitudes (related to coders' observations of shifts in vertical positions of heads, occurrences of horizontal movements of hands and wrist when writing, etc.), clip-level means of Euclidean distances of each keypoint (relating to shifts in groups' total amount of movement), clip-level means of total number of keypoints detected per frame (relating to students opening up their body more/less when interacting/not interacting, leading to potentially increased keypoints being detected), and clip-level means of total number of people detected per frame (relating to teacher more/less likely to walk around during interaction/no-interaction).

For automated feature engineering, we used FeatureTools [22], which combines and calculates new data values based on a relational hierarchy defined by the user. FeatureTools' calculation functions are called primitives, and allows a depth value to be set which enables primitives to be increasingly stacked on top of each other to

Table 2: Summary of trained coders' notes about characteristics they noticed to determine each label.

Code	Characteristics noticed
Interaction (*i*)	Talking with head facing towards someone in group and lips moving, Two people looking at each other actively (e.g., nodding, moving head as they speak), Sharing a worksheet, clearly working from the same document or item, Teacher more frequently walking around/nearby, Exchanging objects, Gesturing towards each other (e.g., pointing, waving)
No interaction (*n*)	Everyone relatively still and looking down at their work, More writing/scribbling movement, Working individually on separate things, Not looking towards any other student group member, Looking away from the group (e.g., at the board or around the class), Overall reduced visibility and lower heads of group members (e.g., leaning down into desk)

create larger feature sets. We used the default set of aggregation-type primitives (count, min, max, mean, skew, standard deviation, sum) along with one transformation-type primitive (percentile), set to a depth of 3, to generate 1,523 features at the clip level (10 seconds). As was the case with manually calculated features, features were individually calculated for each of the four class periods. Some example features generated by FeatureTools were: clip-level minimums of each keypoint's y values, clip-level standard deviation of each keypoint's x values, and clip percentile of the sum of each keypoint's y values.

3.4 Model building

Using the *scikit-learn* Python library [31], we trained a random forest classifier from each of the manual feature sets of the four class periods. We chose random forest due to its suitability in high-dimensional feature spaces without overfitting [6, 15], a key concern with the large number of features considered in this study. We also combined manual features with the automated features and trained additional random forest classifiers from the combined feature set. Feature data had been preprocessed to remove low quality features (invalid values > 5%), which reduced the size of feature sets by 6% to 31% depending on the feature set. We expected automatic features would be difficult to interpret [4], and would likely have led to modest implications for qualitative video coding; thus, we did not create automated features-only classifiers. Models were cross-validated using leave-one-out, where each of the 108 observations were used as the testing set once. Furthermore, the *max_depth* hyperparameter was tuned for each model by changing values 1 to 10 and plotting by accuracy on a validation curve.

4 RESULTS

Here, we discuss the results of the models, including model performance and feature importance analyses.

4.1 Model accuracy

Across the four different class periods, the manual models' accuracies were improved in the combined models (mean manual AUC = .706, mean manual + automated AUC = .778) models. While the base rate of the *interaction* code was not perfectly balanced with the *no interaction* code (except Class A), choosing two equal-length classroom video segments of independent work and small group work led to the labels being relatively balanced. The mean Cohen's κ value in the manual feature models was .453 and combined feature models' mean κ was .555. Model accuracy are compared in Table 3.

4.2 Feature importance analysis

We compared feature importance between the respective manual and combined models. Preprocessing the feature sets in the models variably affected the number of features inputted during model building. For example, Class A had a manual feature set length of 151 and combined feature set length of 1,330, while Class B had 178 and 1,469, respectively. Due to the method in which *scikit-learn* calculates feature importances—in which all the feature importance values sum to 1—we did not directly compare the feature importance values to each other across models. Instead, we compared the top 10% most important features from the folds of each manual models to the top 10% from the folds of the respective combined model.

Results showed that many features important in the manual features model were also important in the combined model, as summarized in Table 4. As high as 90.1% of the most important features from the manual model were found among the important features in the combined model, with a mean of 72.4%. Furthermore, we determined the most frequent keypoints from which models' important features were created, and found that a mean of 5 (50%) of the top 10 (out of 25 total keypoints) most important keypoints overlapped between models. When comparing non-overlapping important keypoints between the models, we found that there was at least one unique body area of features which was important in the combined model but not the manual model. In Class A, these areas were the nose, elbows, and eyes; Class B, neck and wrists; Class C, elbows and shoulders; and in Class D, eyes.

5 DISCUSSION

In the section below, we review our results and discuss them, including potential implications for qualitative coding research.

5.1 Differences in manual vs. automated feature engineering processes

Both manual and combined models had reasonably high accuracy across the respective class periods despite differences in student group location, composition, and orientation. For our first research question, however, we were more interested in examining the differences in the processes of creating manual vs. automated features

Table 3: Accuracy comparison of manual features vs combined (manual + automated) features models.

Video	Interaction base rate	κ (manual)	κ (combined)	AUC (manual)	AUC (combined)
Class A	.500	.352	.444	.676	.722
Class B	.528	.444	.574	.722	.788
Class C	.472	.441	.499	.720	.750
Class D	.491	.574	.704	.704	.852

Table 4: Metrics for feature importance comparison. The top 10% most important features from the manual models were compared to the top 10% from the combined models.

Combined model	Number of features	Manual features in important features	Top 10 key-points overlap	Body area not in manual features
Class A	1330	87.5%	40%	Nose, elbows, eyes
Class B	1469	66.3%	50%	Neck, wrists
Class C	1250	90.1%	60%	Elbows, shoulders
Class D	1112	45.8%	50%	Eyes

related to predicting group interaction. When creating manual features, we were cognizant of the coders' noted characteristics (Table 2). A majority of the manual features were based on values related to shifts in group movement, but other characteristics could not be made into manual features. For example, it was not straightforward to create features around sharing or exchanging objects (e.g., worksheets, book, pencil) because those objects were not detected in the person position data. While our features for movement in the arm keypoints could be one possible proxy for those situations, it was not possible to create a direct feature. Lips moving, or talking, was arguably the most frequent and defining characteristic of interaction, but those keypoint values were not available in our data. Despite these limitations, the model created from manual features worked well.

The time and effort to create automated feature engineering process using FeatureTools was notably less than the manual process. FeatureTools was able to generate 1,523 features with little researcher time. The resulting feature names, however, were often difficult to interpret, since FeatureTools combined column names with the raw calculations carried out on the data. In many cases, features created by FeatureTools were nonsensical, such as applying percentile calculations to the rows of binary column data (e.g., keypoint detected or not detected), or the counts of the sums of keypoint y values. The model built from the combined feature set had improved accuracy compared to the manual model, perhaps due to FeatureTools' features being able to capture and represent trends in the data overlooked by our manual features. Automated feature engineering tools' tendency to create models of high accuracy but lowered interpretability of features has been previously explored [4], and our study mirrors those findings.

5.2 Automated methods for informing manual feature engineering

Our second research question was concerned with how automated feature engineering methods might inform manual feature engineering methods. Feature importance comparisons between the models showed that in all class sessions, manual features were consistently included among the most important features in the combined model. When understood together with the high accuracy of manual feature models, this may show that the largely group-movement-based manual features were more parsimonious in effectively capturing characteristics related to group interaction despite having a comparatively smaller number of features.

We also found that there were overlaps between the top 10 most important keypoints for predicting group interaction between the two models. This meant that many of the same keypoint values among 25 were used to create important features in each model. The overlaps may be unsurprising as both models were likely able to find that interactions were largely characterized by changes in specific areas of the body. These overlapping keypoints in the two models, however, could be used to ascertain the relative usefulness of certain keypoints over others. This information could be used to help the researcher strategize which keypoint data should be perhaps receive more attention when manually creating features. Similarly, the important features and body areas found exclusively in the combined models' set of important features may have potential value for manual feature engineering. Across the four classroom sessions, the combined model found keypoints from unique upper body areas (i.e., areas not found among important body areas in the manual model) to be important for predicting student group interaction. Manual features could then be designed with this information (Table 4), such as brainstorming and creating specific features related to the nose, elbow, and eye for Class A's model. This could result in a more comprehensive feature set that attempts to integrate the potentially less noticeable constructs related to group interaction for that particular video data.

5.3 Implications for qualitative video coding

Feature importance analyses such as those in our study could be carried out as a preliminary pilot study to allow qualitative coders to confirm and expand their observations of relevant behavior characteristics for code definitions before coding a larger set of data. In our study, qualitative coders had noted that group interactions

were largely characterized by changes in movement, such as moving lips or making gestures. Our results showed that primarily movement-based features in the manual feature set were effective in predicting group interaction. The presence of important manual features among the important features in combined models could perhaps be used by researchers who are coding for group interaction to strongly confirm the importance of movement as relating to group interaction. Additionally, based on how the respective combined models' accuracy were even further improved from the manual models, the added meaning captured by the automated features could inform qualitative video coders about potentially unexpected, yet relevant body areas to group interaction. For example, by considering such information in the last column of Table 4, coders can be informed to more closely examine the nose, elbows, and eyes of participants in Class A if they were not already doing so.

6 CONCLUSION AND FUTURE WORK

In this paper, we were motivated to explore how automated methods could inform human perspectives around analyzing qualitative classroom video data. Based on our observations of four middle school and high school mathematics class video data, we labeled group interaction codes, and carried out expert-informed manual feature engineering and automated feature engineering (using FeatureTools) from tracked positional data. Our results highlighted differences in the modeling processes, such as improved model accuracy when adding automated approaches (mean manual models AUC = .706, mean combined models AUC = .778) despite a decrease in overall feature interpretability, more directed feature brainstorming in manual feature engineering, and shortened process time and increased number of automated features (1,523 vs. 178). A large proportion of important features from manual feature set models were also important in combined feature set models (mean = 72.4%). We also discussed our methods and results in terms of how qualitative video researchers may use similar approaches to inform their qualitative video coding processes.

In terms of limitations, our study used just four classroom videos to explore our approach as we wanted to observe substantial sessions of continuous small group work and individual work for each class session. Thus, were not able to reach a large number of group interaction labels with 104 labels coded at 10-second clips, for a total of 416 labels. This most likely reduced observations of unique student interactions, and our results may not be representative of general small group interactions. Future work could determine the suitability of integrating the unique body areas found to be important by the combined models for informing qualitative coding researchers, exploring a wider variety of student classroom interactions beyond small group interaction, such as constructs around teacher–student interactions, and explore additional automated feature engineering tools. We are currently investigating some practical applications of this work, such as how qualitative video analysts would perceive and utilize automatically-filtered video clips from a class period. Classroom video data are rich and detailed sources for mining insights about student learning. Our paper highlights how automated methods may have the potential to augment researchers' perspectives on making sense of complex, unstructured classroom video data.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation (DRL-1920796). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] Karan Ahuja, Dohyun Kim, Franceska Xhakaj, Virag Varga, Anne Xie, Stanley Zhang, Jay Eric Townsend, Chris Harrison, Amy Ogan, and Yuvraj Agarwal. 2019. EduSense: Practical classroom sensing at Scale. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–26.

[2] Alicia C Alonzo, Mareike Kobarg, and Tina Seidel. 2012. Pedagogical content knowledge as reflected in teacher–student interactions: Analysis of two video cases. *Journal of Research in Science Teaching* 49, 10 (2012), 1211–1239.

[3] Robert Bogdan and Sari Knopp Biklen. 1997. *Qualitative Research for Education*. Allyn & Bacon, Boston, MA.

[4] Nigel Bosch. 2021. AutoML feature engineering for student modeling yields high accuracy, but limited interpretability. *Journal of Educational Data Mining* 13, 2 (2021), 55–79.

[5] Nigel Bosch, Sidney K. D'Mello, Jaclyn Ocumpaugh, Ryan S. Baker, and Valerie Shute. 2016. Using video to automatically detect learner affect in computer-enabled classrooms. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 6, 2 (2016), 1–26.

[6] Leo Breiman. 2001. Random forests. *Machine Learning* 45, 1 (2001), 5–32.

[7] Raymond Brown and Peter Renshaw. 2006. Positioning students as actors and authors: A chronotopic analysis of collaborative learning activities. *Mind, Culture, and Activity* 13, 3 (2006), 247–259.

[8] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime multi-person 2D pose estimation using part affinity fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 7291–7299.

[9] Maximilian Christ, Nils Braun, Julius Neuffer, and Andreas W. Kempa-Liehr. 2018. Time series feature extraction on basis of scalable hypothesis tests (TSFRESH–a Python package). *Neurocomputing* 307 (2018), 72–77.

[10] Fabio Dovigo. 2020. Through the eyes of inclusion: An evaluation of video analysis as a reflective tool for student teachers within special education. *European Journal of Teacher Education* 43, 1 (2020), 110–126.

[11] Fred C. Feitler, William Wiener, and Arthur Blumberg. 1970. The relationship between interpersonal relations orientations and preferred classroom physical settings. (1970), 18 pages.

[12] Nan Gao, Wei Shao, Mohammad Saiedur Rahaman, and Flora D. Salim. 2020. n-gage: Predicting in-class emotional, behavioural and cognitive engagement in the wild. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 3 (2020), 1–26.

[13] Isabelle Guyon, Steve Gunn, Masoud Nikravesh, and Lofti A. Zadeh. 2008. *Feature Extraction: Foundations and Applications*. Vol. 207. Springer.

[14] Zahra Hazari, Cheryl Cass, and Carrie Beattie. 2015. Obscuring power structures in the physics classroom: Linking teacher positioning, student engagement, and physics identity development. *Journal of Research in Science Teaching* 52, 6 (2015), 735–762.

[15] Tin Kam Ho. 1995. Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, Vol. 1. IEEE, 278–282.

[16] Fiona Hollands and Ipek Bakir. 2015. *Efficiency of automated detectors of learner engagement and affect compared with traditional observation methods*. Technical Report. Center for Benefit-Cost Studies of Education, Teachers College, Columbia University, New York, NY. 37 pages. https://repository.upenn.edu/cbcse/4

[17] Franziska Horn, Robert Pack, and Michael Rieger. 2019. The autofeat Python library for automated feature engineering and selection. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, Cham, CH, 111–120.

[18] Paul Hur and Nigel Bosch. 2022. Tracking individuals in classroom videos via post-processing OpenPose data. In *LAK22: 12th International Learning Analytics and Knowledge Conference*. ACM, New York, NY, 465–471.

[19] Paul Hur, Nigel Bosch, Luc Paquette, and Emma Mercier. 2020. Harbingers of collaboration? The role of early-class behaviors in predicting collaborative problem solving. In *Proceedings of the 13th International Conference on Educational Data Mining*. International Educational Data Mining Society, 104–114.

[20] Paul Samuel Ignacio, Christopher Dunstan, Esteban Escobar, Luke Trujillo, and David Uminsky. 2019. Classification of single-lead electrocardiograms: TDA

informed machine learning. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. IEEE, 1241–1246.

[21] Yang Jiang, Nigel Bosch, Ryan S. Baker, Luc Paquette, Jaclyn Ocumpaugh, Juliana Ma Andres, L. Alexandra, Allison L. Moore, and Gautam Biswas. 2018. Expert feature-engineering vs. deep neural networks: which is better for sensor-free affect detection?. In *Proceedings of the 19th International Conference on Artificial Intelligence in Education*. Springer, Cham, CH, 198–211.

[22] James Max Kanter and Kalyan Veeramachaneni. 2015. Deep feature synthesis: Towards automating data science endeavors. In *2015 IEEE international conference on data science and advanced analytics (DSAA)*. IEEE, 1–10.

[23] Christina Krist. 2020. Examining how classroom communities developed practice-based epistemologies for science through analysis of longitudinal video data. *Journal of Educational Psychology* 112, 3 (2020), 420.

[24] Marjan Laal and Seyed Mohammad Ghodsi. 2012. Benefits of collaborative learning. *Procedia - Social and Behavioral Sciences* 31 (2012), 486–490.

[25] Kevin M Leander and Margery D Osborne. 2008. Complex positioning: Teachers as agents of curricular and pedagogical reform. *Journal of Curriculum Studies* 40, 1 (2008), 23–46.

[26] Fei Victor Lim, Kay L O'Halloran, and Alexey Podlasov. 2012. Spatial pedagogy: Mapping meanings in the use of classroom space. *Cambridge Journal of Education* 42, 2 (2012), 235–251.

[27] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172* (2019).

[28] Roberto Martinez-Maldonado, Vanessa Echeverria, Jurgen Schulte, Antonette Shibani, Katerina Mangaroska, and Simon Buckingham Shum. 2020. Moodoo: Indoor positioning analytics for characterising classroom teaching. In *International Conference on Artificial Intelligence in Education*. Springer, Cham, CH, 360–373.

[29] Selina McCoy and Aoife M Lynam. 2021. Video-based self-reflection among pre-service teachers in Ireland: A qualitative study. *Education and Information Technologies* 26, 1 (2021), 921–944.

[30] Luc Paquette, Nigel Bosch, Emma Mercier, Jiyoon Jung, Saadeddine Shehab, and Yurui Tong. 2018. Matching data-driven models of group interactions to video analysis of collaborative problem solving on tablet computers. International Society of the Learning Sciences, Inc.

[31] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[32] Mar Pérez-Sanagustín, Patricia Santos, Davinia Hernández-Leo, and Josep Blat. 2012. 4SPPIces: A case study of factors in a scripted collaborative-learning blended course across spatial locations. *International Journal of Computer-Supported Collaborative Learning* 7, 3 (2012), 443–465.

[33] Penelope L. Peterson and Susan R. Swing. 1985. Students' cognitions as mediators of the effectiveness of small-group learning. *Journal of Educational Psychology* 77, 3 (1985), 299.

[34] Mar Saneiro, Olga C. Santos, Sergio Salmeron-Majadas, and Jesus G. Boticario. 2014. Towards emotion detection in educational scenarios from facial expressions and body movements through multimodal approaches. *The Scientific World Journal* 2014 (2014), 14 pages.

[35] Carol S. Weinstein. 1981. Classroom design as an external condition for learning. *Educational Technology* 21, 8 (1981), 12–19.

[36] Lixiang Yan, Roberto Martinez-Maldonado, Beatriz Gallo Cordoba, Joanne Deppeler, Deborah Corrigan, Gloria Fernandez Nieto, and Dragan Gasevic. 2021. Footprints at school: Modelling in-class social dynamics from students' physical positioning traces. In *LAK21: 11th International Learning Analytics and Knowledge Conference*. 43–54.

[37] Jiaoping Zhang, Hsiang Sing Naik, Teshale Assefa, Soumik Sarkar, R.V. Chowda Reddy, Arti Singh, Baskar Ganapathysubramanian, and Asheesh K. Singh. 2017. Computer vision and machine learning for robust phenotyping in genome-wide studies. *Scientific Reports* 7, 1 (2017), 1–11.

[38] Heather Toomey Zimmerman and Susan M. Land. 2022. Supporting children's place-based observations and explanations using collaboration scripts while learning-on-the-move outdoors. *International Journal of Computer-Supported Collaborative Learning* 17, 1 (2022), 107–134.