Learning Deep Social Interactions to Identify Positive Climate

Przemek Gardias [1]

Abstract

Recent work contributing to the automation of CLASS positive climate (PC) classification has demonstrated success in using deep models to learn classroom climate by modeling the scene of a classroom as a social network graph. We theorize that by tracking participants across a social graph, we can attain higher CLASS prediction accuracy compared to previous work which ignored identities of students [1]. We discuss initial experimentation on simulated classroom observations to evaluate the effect of tracking participant nodes to identify interactions, and outline the performance improvements of this method considering our intuition behind PC annotation. Furthermore, we suggest methods of tracking participants required to construct such a data transformation and propose several experiments on an authentic dataset of preschool classroom interactions (UVA Toddler).

1. Introduction

A widely-used classroom observation protocol used by educational researchers is the Classroom Assessment Scoring System, CLASS [2], which requires trained human annotators to examine the state of the classroom and students for qualities that exhibit social, organizational, and instructional support. CLASS is a valuable tool for teachers and educational researchers–but human annotation is slow, expensive, and requires weeks to months of training. Typical CLASS annotation sessions require annotators to examine specific characteristics of the states, actions, and interactions among the students and teachers during either live observation or video recordings. A 2017 study from Chile found costs of individual classroom annotation to be in the order of $100 [3]. The magnitude of these costs makes it difficult to provide raw data for teacher feedback, a useful mechanisms for providing resources for teachers to adjust their teaching methods to effectively support all students. We seek to improve on initial work towards learning classroom climate classification by focusing on CLASS-defined social interactions in a weighted social graph representation of the classroom scene. By definition we focus on identifying positive interactions between student-teacher pairs to verify social support structures within the classroom environment.

[1]Worcester Polytechnic Institute. Correspondence to: Przemek Gardias <pmgardias@wpi.edu>.

2. Prior Work

The first work towards automating aspects of CLASS annotation made strides to estimate 3 minute clips of classroom observation videos which were most relevant to CLASS annotators to code manually [4]. However, more recently, there have been a number of efforts to analyze the dynamics of a classroom, some focused on an aggregate measure, such as [1] and [5], while others focused on individual students [6]. Although there are many approaches for harnessing deep learning for measuring educational metrics such as CLASS, many prior efforts focus on analyzing student engagement and emotions [7]. While there are many different approaches to labeling classroom observations using CLASS metrics, we aim to distinguish short video clips with high or low positive climate (PC).

2.1. Graph Convolution for Social Network Graphs

Graph convolution networks (GCNs) are a category of recent deep learning architecture applicable for problems modeled as graphs. Although there are two notably different approaches to applying a convolution transform on a graph data structure, spectral and spatial graph convolutions, they are similar in their outcome: label information is smoothed out over the graph via a form of explicit graph-based regularization [8]. This transform can be used for node or graph classification tasks, where it is useful to transfer information from neighboring nodes, in a manner similar to a 2D convolution layer, while considering the topology of the graph.

Recent work in identifying classroom climate, ACORN [1], achieves significant inter-coder reliability results with respect to expert labels using a multi-modal deep learning methods ensemble. Some of the experimentation includes GCNs, attention layers, and a 3-layer bidirectional LSTM to aggregate temporal features. Additionally, with

an experiment using a uniform normalized Laplacian matrix which contains identity matrix I and where all other weights are $\frac{1}{d}$, such that the graph is a clique, ACORN establishes that graph topology, or *who is where and interacting with whom when* is important for estimating classroom PC, achieving an average of $AUC = 0.70$ across 10-folds.

Traditional deep learning methods, particularly convolutional neural networks (CNN) have been shown to perform poorly on data with underlying graph structures, such as social network graphs. Some methods explore extending CNN components to graphs with graph signal processing (GSP) [9]. Graphs have been demonstrated to be perfect for capturing node interactions, especially on non-Euclidean data domains. A recent application of GCNs on a problem modeled as a spatio-temporal social network graph achieves state of the art results and shows the model is able to capture behavior expected in humans [10]. The same work notably achieves these results with significantly less parameters are and a fraction of the training data used by previously comparable methods which did not model the scene as a graph.

2.2. Machine Perception for Classroom Observation

There are recent methods which leverage computer vision for automating aspects of CLASS that identify relevant segments of classroom video important for coders rather than predicting a label [4]. Other methods use dedicated hardware to unify a multitude of contemporary machine learning techniques [6]. Due to the requirements of our approach, we consider off-the-shelf object recognition and tracking tools. Visual perception utilities such as OpenPose [11] and OpenFace [12] offer similar utility as cloud services–and are successful applied as low-level feature engines, processed to estimate higher-level features. A concern when applying these tools is our niche classroom environment recognition task. Often, researchers build bespoke perception systems specific to student learning environments [13]. Similar toolkits are available for object tracking [14], but only some address multi-object tracking (MOT) [15] [16]. These techniques are applied as part of our scene to graph pipeline shown in Figure 1.

3. Proposed Research

We theorize that if we use social graphs representations of our classroom observations and apply a spatial graph convolution layer we can capture key interactions between participants which are central to distinguishing between high and low PC. We propose a series of experiments to compare the ability to learn CLASS climate by tracking participants in classroom videos, and demonstrate this on experiments with increasing complexity of node interactions. We define a social network graph from each frame of classroom observations, and use object recognition to generate each of the node's features, a 4×1 column vector consisting of smiling, anger, sadness, and $Pr(student)$ (equivalent to $1 - Pr(teacher)$) features [1]. Our social network graph follows a standard graph data structure: n nodes in a given graph $G_t = (V_t, A_t)$ with v features per node, such that $V \in \mathbb{R}^{n \times v}$. Our adjacency matrix $A \in \mathbb{R}^{n \times n}$ is weighted with the inverse pixel distance between nodes. Given our weighted graph structure, where all edges are non-zero, such that $|A| = \frac{|V| \times (|V|-1)}{2}$, *all* nodes are convolved in each spatial graph convolution layer; we have to take care to avoid over-convolving towards a fully entropic graph state. Inspired by the simple formulaic depiction of feed-forward neural network, which omits the bias term, we can define our graph convolution layer as follows: for layer l, our spatial graph convolution layer activation matrix $H_t^{(l)}$ is

$$H_t^{(0)} = V_t$$
$$H_t^{(l+1)} = \sigma(L_t H_t^{(l)} W^{(l)})$$

where our symmetric normalized Laplacian L_t is

$$L_t = I_t - D_t^{-\frac{1}{2}} A_t D_t^{-\frac{1}{2}},$$

$W^{(l)}$ is the convolution kernel weights of the given layer, and σ is our non-linear activation function ReLU. We follow each graph convolution layer with layer normalization to address the exploding gradient problem. Using this approach for our graph convolution task we define a tracking and tracking-adverse pair of models for PC classification, shown in Figure 1.

3.1. Research Questions

We propose a number of questions to focus our experiments on. We aim to address these points to verify the ability of our model: can we learn social interaction by tracking participants in a social network graph, and does reflecting our understanding of how CLASS climate is labeled by human annotators in the network architecture improve this ability?

- Does tracking participants in a social scene help in improving performance of PC classification, compared to a tracking-free approach which does not consider the identities of students and teachers?
- Can we, using node attention mechanisms in our graph convolution layers, identify node interactions? How can we verify our model is learning these interactions rather than fitting to intrinsic trends in our data? Are there mechanisms we can use to validate our expected behavior when training on simulated data?
- Are there architecture choices with which we can make learning interactions more general, considering

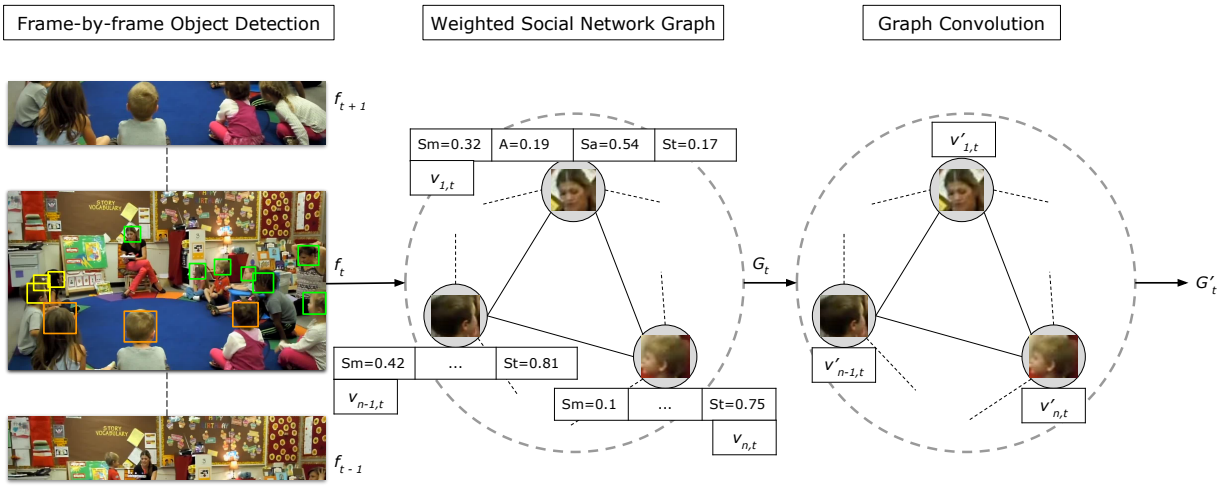


Figure 1. An example sequence of frames f_{t-1}, f_t, f_{t+1} is used to show the creation of a weighted social network graph $G_t = (V_t, A_t)$ from frame f_t, where each node $n_i \in V_t$ contains feature vector $v_{n,t}$, consisting of emotion and age information as proposed in [1].

our real-world data? A deeper network approach–applying convolution-based feature mapping–is difficult to train on our small dataset, so we use off-the-shelf tools for our feature vector generation. Given this consideration, can we somehow use a highly abstract feature matrix (e.g. facial pixels) to improve the accuracy and generalization of our model? Can we apply other CLASS-coded datasets as our unseen test set to verify generalization improvements? Are we able to compare these approaches and identify a training set size at which a deeper learning approach may fare better?

- Can we exploit an ensemble model earlier in our graph construction pipeline to create a more meaningful feature vector with which we can improve our accuracy by minimizing our dependency on a single expression classification tool?

- How accurately can we track nodes? How will our ability to track affect our results: given a tracking accuracy across frames, can we estimate how well we expect to do?

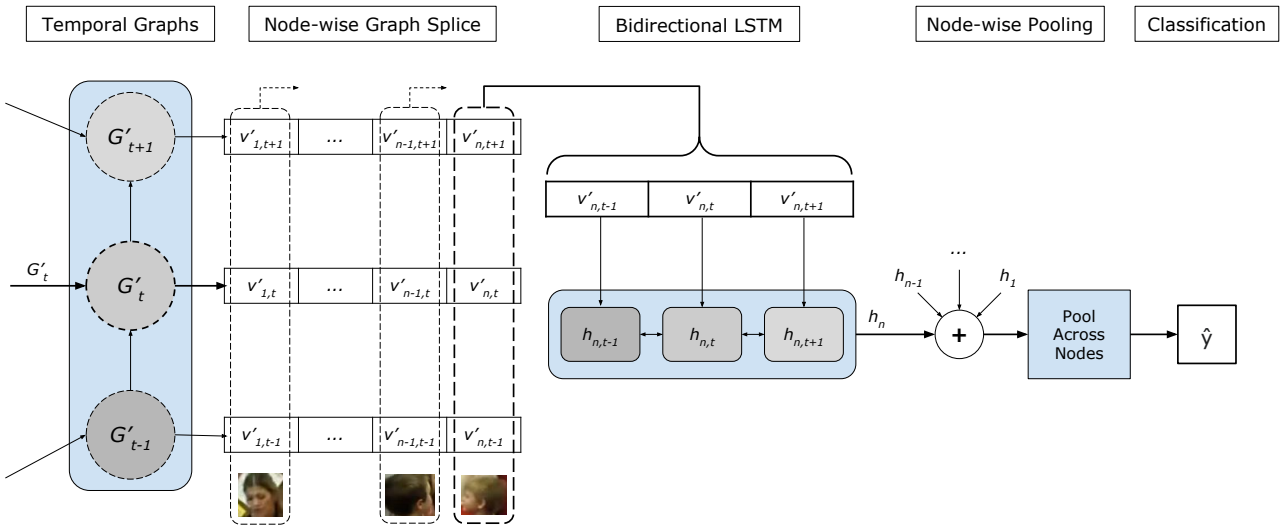
3.2. Model Proposals

We introduce several shallow models for simple node tracking which are able to quickly learn simple feature-based thresholds required for classification. Additionally, we propose several models which include similar architecture choices, although generally aim to be more broadly applicable and therefore successful on our real world dataset as compared to simulated problems.

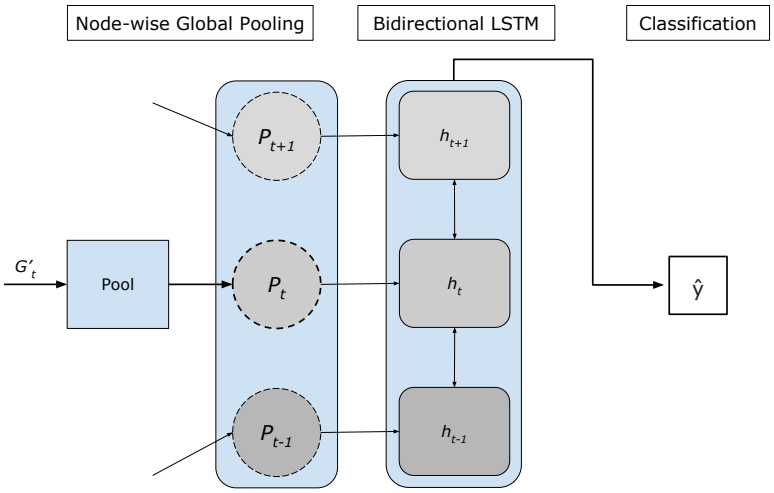
- A graph convolution and shared-weight LSTM architecture which processes node-wise time series and then congregates the individual hidden states via a graph pooling mechanism, to then classify climate. Figure 2 compares this architecture to the equivalent network which ignores participant identities. This approach aims to capture the complexity of participant interactions as a means of PC identification further than the simulations discussed in Sections 3.5 & 3.3. Although comparing increasing complexity ground truth labeling functions can help identify performance of the network in learning these defined trends, we specifically aim to generalize our approach by minimizing feature engineering efforts on simulated data. Success found in these efforts would be transferable to work on the UVA Toddler dataset.

- We aim to evaluate techniques shown in ACORN [1] to improve correlation, such as attention-based graph convolution and pooling, and experiment with combining these methods with our approach, using mechanisms which have shown promise on graph classification tasks, such as Gated Graph Neural Networks (GGNN) [17].

Furthermore, we consider the methods with which we can track nodes between temporal graphs G_t, G_{t+1}: traditional object tracking methods such as GOTURN [14], or trying to exploit our knowledge of limited nodes with using unsupervised node embedding network, such as *node2vec* [18]. Comparing different levels of abstraction of our features for embedding may be useful if seeking to improve consistency of tracking, but nonetheless we aim to establish a node-wise sorted vertex set V for all frames to fulfill the assumptions held by our tracking-based network.



a) Tracking network



b) Tracking-free network

Figure 2. Architecture diagrams depicting our proposed a) tracking-based network and b) tracking-free network. Both models use classroom observation video as input–then diverge in node tracking assumptions. Model a) includes a graph convolution layer for each time step t, after which the convolved feature vectors $v'_{n,t}$ are concatenated temporally to once again represent a spatio-temporal social network. We splice the vertex set V_t node-wise such that each tracked node has an individual time series, and then aggregate using a bidirectional LSTM which is used for each node time series but shares weights across all nodes. The then pool across the output hidden states such that we retain information from all nodes. On the other hand, model b) does not assume tracking information is retained across the time steps of the temporal social network graph and includes a global pooling layer across nodes such that $pool(G_t) \in \mathbb{R}^{v \times 1}$, followed with a bidirectional LSTM layer. Both models include a final fully-connected layer for binary classification.

3.3. Simulated Interactions

To evaluate the design choices of our architecture model, we first define a tracking problem and corresponding dataset. We use a plausible classroom scenario, where tracking the state of nodes is directly relevant to the PC label, and construct randomly generated time series of length t seconds for n nodes. This system of labeling seeks to replicate the process of identifying key classroom moments within the larger duration of the video.

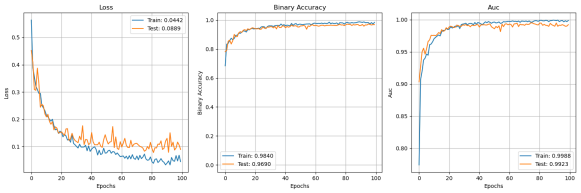
We work forward, generating our input $x \in \mathbb{R}^{n \times t}$ by simply sampling $x_{n,t} \sim \mathcal{U}(0,1)$ and evaluating for y as follows: for each time step t, each feature $x_{n,t}$ in our simulated time series must pass a threshold, an activation which must be first met independent of other nodes and then passed by another threshold of nodes. For our purposes, these thresholds are 50%. Therefore, we introduce a inter-node requirement that is unable to be captured with an architecture that does not perform node-wise processing. We expect a higher complexity dataset, with higher dimension features and a more complex ground truth label function, would perform significantly differently and comparisons of tracking methods would be difficult to evaluate consistently.

3.4. Emulating CLASS

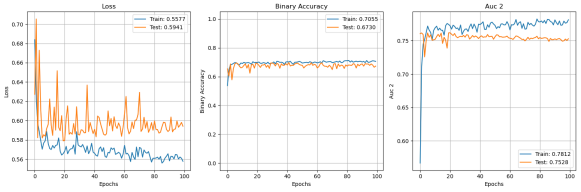
To continue our experimentation, we add two heavily problem altering considerations for determining our ground truth: proximity-based interactions and obfuscation with additional features. Our input expands to another dimension to contain f features, which we again sample from $\mathcal{U}(0,1)$. Teachers and students are ensured to always be distinguishable by constraining the sample distribution to $(0, 0.5)$ and $(0.5, 1)$, respectively. In addition to features, we simulate participant movement across a frame by first initializing participants across our standard scene of size $w \times h$ and sampling movement vectors of each participant:

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \gamma * \begin{bmatrix} \mathcal{N}(0, w) \\ \mathcal{N}(0, h) \end{bmatrix},$$

where γ is a small weight to limit movement. We clip the final positions of participants such that it cannot exceed the bounds of our simulated frame, and prevent sparse graph representations due to occlusion or exit of scene events, which are common in real world data. In addition to these changes, we adjust our logic of evaluating y by defining a proximity threshold, a proportion of the diagonal of the frame, to count a student-teacher interaction as positive. Although not shown in the relevant results discussed in Section 3.5, we explore additional complexity such as emulating sadness and anger features and implementing similar threshold-based logic to identify meaningful interaction only when participants are not overtly negative in expression.



a) Tracking model



b) Tracking-free model

Figure 3. Training results of a simple 1-feature simulation for a) the tracking-based network compared to b) the tracking-adverse model. Both models were trained for 100 epochs, using $lr = 1e-2$, with our dataset simulation configured to $n = 4$ nodes and $t = 10$ timesteps. Such a configuration allows the models to quickly fit the dataset by learning node interaction requirements we defined in our ground truth label function.

3.5. Initial Results

We apply a straightforward architecture for classification, where a bidirectional LSTM layer is either a) followed by a pooling layer, as in the tracking-based network, or b) preceded by a pooling layer, as is implemented in the tracking-adverse network. We examine the effect which node-level data loss contributes to the network's ability to fit to a correlation defined for each individual time series by comparing test classification accuracy.

As shown in Figure 3, we identify a significant accuracy

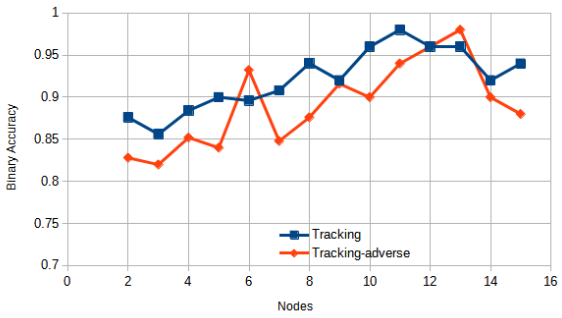
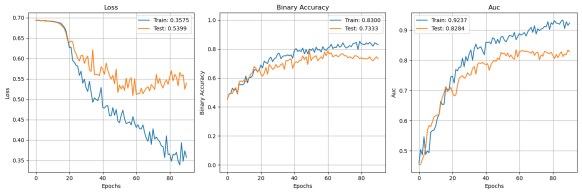
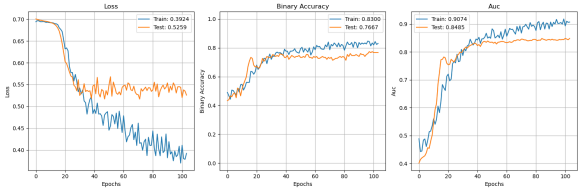


Figure 4. Tracking vs. tracking-free test binary accuracy for n nodes. Models were trained for 100 epochs, with early stopping and optimal weight restoration to maximize validation set metrics for each simulation configuration.



a) Tracking model



b) Tracking-free model

Figure 5. Training results of a simulation for a) the tracking-based network compared to b) the tracking-adverse model. Both models were trained for 100 epochs, using $lr = 1e - 3$, with our dataset simulation configured to $n = 22$ nodes and $t = 10$ seconds.

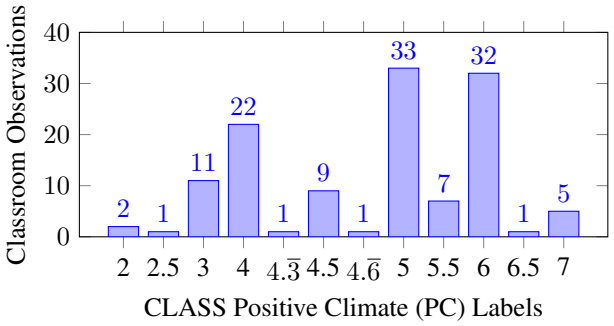


Figure 6. Distribution of positive climate labels of 125 classroom observation videos in the UVA Toddler dataset. Given the non-uniform distribution of PC labels, it is clear why a regression approach is necessary, in combination with efforts to counter the effects of an unbalanced dataset on model fitting.

improvement when tracking participants, primarily due to information loss required for the interaction learning we seek to accomplish. These results affirm two key points: our tracking-based architecture seem to be able to capture the node interactions, and the tracking-adverse model is able to perform seemingly well even in the absence of learning interactions. Identifying the performance difference provides us a baseline comparison differential of node-level learning, which is especially useful in the context of real-world data. This supports further investigation of expanding our simulation to include more participants additionally validated by Figure 4, which similarly presents a performance difference in a trend of increasing accuracy given increasingly larger graphs. We can hypothesize, following the assumption that CLASS climate labels are based upon node interactions, that given a sufficiently deep node feature capture method, we can identify a method which employs graph convolution for social network feature propagation and tracking-based time series processing, that captures this interaction sufficiently enough to perform better than a tracking-adverse network. In this way, we seek to identify a method which is able to understand the causal mechanism of how the node-wise interactions result in CLASS climate.

Initial results of further experimentation on the efficacy of the proposed architecture on our problem, by effectively further obfuscation of the logic in our ground truth label function, are shown in Figure 5. By increasing the simulated participant features available, and expanding the logic of our ground truth labels to account for complex interactions which are reliant on more of the available features, we create a significantly more difficult to capture differentiation between labels. Although most notably the differences between model accuracy are not clearly identifiable, this lack of an accuracy gap may be indicative of our models ability to capture very nuanced interactions which are reliant on several feature requirements in tandem. This introduces an opportunity for node attention mechanisms to identify key neighboring nodes for our graph convolution steps, which we expect to be necessary for success considering the initial findings of [1]. Furthermore, we do not concede ability to achieve success on real world data, given the difficulty to represent the true relationship between social interactions and PC labels as exists in the UVA Toddler dataset.

3.6. Datasets

The University of Virginia (UVA) Toddler dataset consists of 192 CLASS-coded videos, each approximately 45-60 minutes long. The videos are from 61 early childhood care centers, where the students are toddlers 2-3 years old. All videos were recorded from classrooms in a Mid-Atlantic state of the USA. For our purpose, each of the videos is split into short clips with high or low PC. To lessen the effect of erroneous social network construction, and to simplify our tracking problem, we cap the number of face detections in each frame to 22, the maximum number of participants in any of the UVA Toddler classroom videos.

In a similar fashion as ACORN [1], we can investigate our ability to generalize to elementary and middle school students, as well as compare climate classification accuracy using the Measures of Effective Training (MET) dataset, which contains thousands of videos and is similarly CLASS-coded.

4. Timeline

Objective	Dates
Evaluation on UVA Toddler	Aug. 31 - Oct. 31
Tracking Participants	Nov. 1 - Dec. 16
Attention & GGNN Experimentation	Jan. 12 - Feb. 19
Thesis (Draft)	Feb. 20 - Mar. 18
Thesis (Final)	Mar. 19 - Apr. 21

References

[1] Anand Ramakrishnan, Brian Zylich, Erin Ottmar, Jennifer LoCasale-Crouch, and Jacob Whitehill. Toward automated classroom observation: Multimodal machine learning to estimate class positive climate and negative climate. *arXiv preprint arXiv:2005.09525*, 2020.

[2] Robert C Pianta, Karen M La Paro, and Bridget K Hamre. *Classroom Assessment Scoring System™: Manual K-3*. Paul H Brookes Publishing, 2008.

[3] World Bank. The classroom assessment scoring system (class), 2017.

[4] Qifeng Qiao and Peter A Beling. Classroom video assessment and retrieval via multiple instance learning. In *International Conference on Artificial Intelligence in Education*, pages 272–279. Springer, 2011.

[5] Tsung-Yen Yang, Ryan S Baker, Christoph Studer, Neil Heffernan, and Andrew S Lan. Active learning for student affect detection. *International Educational Data Mining Society*, 2019.

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

[7] Kenneth Holstein, Bruce M McLaren, and Vincent Aleven. Student learning benefits of a mixed-reality teacher awareness tool in ai-enhanced classrooms. In *International conference on artificial intelligence in education*, pages 154–168. Springer, 2018.

[8] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[9] Mark Cheung, John Shi, Oren Wright, Lavender Y Jiang, Xujin Liu, and José MF Moura. Graph signal processing and deep learning: Convolution, pooling, and topology. *arXiv preprint arXiv:2008.01247*, 2020.

[10] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14424–14432, 2020.

[11] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017.

[12] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. Openface: an open source facial behavior analysis toolkit. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–10. IEEE, 2016.

[13] Jacob Whitehill, Zewelanji Serpell, Yi-Ching Lin, Aysha Foster, and Javier R Movellan. The faces of engagement: Automatic recognition of student engagementfrom facial expressions. *IEEE Transactions on Affective Computing*, 5(1):86–98, 2014.

[14] David Held, Sebastian Thrun, and Silvio Savarese. Learning to track at 100 fps with deep regression networks. In *European Conference on Computer Vision*, pages 749–765. Springer, 2016.

[15] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88, 2020.

[16] Zhongdao Wang, Liang Zheng, Yixuan Liu, and Shengjin Wang. Towards real-time multi-object tracking. *arXiv preprint arXiv:1909.12605*, 2019.

[17] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*, 2015.

[18] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.