# How Can We Use Machine Learning to Improve The Learning Experience On the Active Learning Forum (ALF)?

Mohamed Gad, Quang Tran, Tanha Kate, Yuhao Chen

FE51 Spring 2018

**TABLE OF CONTENTS**

## 1. Introduction

As a team we were very excited to receive our Final Project question and thrilled at the prospect of learning more about Machine Learning (ML) techniques. However, we got overwhelmed later by the massive amount of information on the topic. Anxiety resulted from the thought that we might not have the time to learn and implement a novel method to meet the project deadline. We finally decided to look into Convolutional Neural Networks (CNNs) by downloading a recommended paper, which triggered our curiosity with its eye catching visuals and motivated us to explore further. The curiosity quickly turned into excitement when we knew more about the complexity of emotions in a learning environment. This motivated us to slog through the technical details, which led to some intense emotions - confusion and frustration when things did not make sense, despair when we almost gave up, and eventually, delight when we made progress. Some more emotions occurred during the writing and revision cycles. Finally, we were done. We felt contentment, relief, and pride.[1]

This example illustrates an undercurrent of emotion throughout the learning process. In this paper, we will summarize secondary-literature findings concerning the importance of emotion in an academic setting and prior attempts at facial recognition and introduce a development plan for fully automatic real-time emotion recognition using CNNs, with details of our algorithm and a preview of the interface design. We will conclude with a discussion of future plans.

### *1.1 Background*

Emotions, even when they aren't consciously experienced, exist and influence cognition (Ohman & Soares, 1994 as cited in D'Mello, 2017) in academic settings. Many psychologists previously conceptualized academic thinking as primarily a cognitive activity, relatively free from emotion (Brown et al. 1983). However, modern social scientists concede that emotions are not merely incidental; they are functional and have evolved over time (Darwin, 1872; Tracy, 2014 as cited in D'Mello, 2017):

[1] **#cognitivepersuasion**: We used the technique of narrative paradigm, which states that storytelling can permeate a receiver's consciousness and influence cognitive processes. Beginning with details in the narration on a personal level before presenting any scientific claims makes the topic more approachable, understandable and thus make the readers more open to accept the arguments about the importance of emotional recognition technology.

| Signalling | Evaluative | Modulation |
|---|---|---|
| Highlight problems with knowledge (confusion), problems with stimulation (boredom), concerns with impending performance (anxiety) and challenges that cannot be easily surpassed (frustration) (Schwarz, 2012). | Serves as the currency by which people appraise an event in terms of its value, goal relevance and goal congruence) (Izard, 2010). | Constrain or expand cognitive focus with negative emotions engendering narrow, bottom-up and focused modes of processing (constrained focus) (Barth & Funke, 2010; Schwarz, 2012) in comparison to positive emotions, which facilitate broader, top-down, generative processing (expanded focus) (Friedrickson et al., 2005; Isen, 2008 as cited in D'Mello, 2017) |

**Table 1.** Functional effects of emotions

Emotions are conceptual entities which arise from brain-body-environment interactions but cannot be understood by looking at either component (D'Mello, 2017). They emerge (Lewis, 2005 as cited in D'Mello, 2017) when organism-environment interactions trigger changes across multiple time scales and at multiple levels.

'Academic emotions' are those experienced by a typical student and its effects on motivational, cognitive, and cognitive-behavioral engagement (Pekrun and Linnenbrink, 2012). Academic emotions are grouped into four categories:

- *Achievement emotions* (contentment, anxiety and frustration) are linked to learning activities (homework, taking a test) and outcomes (success, failure).
- *Topic emotions* are aligned with the learning content
- *Social emotions* such as pride and jealousy occur in social contexts.
- *Epistemic emotions* arise from cognitive processing (e.g., surprise when novelty is encountered.)

This paper will deal with achievement emotions since various measures exist to operationalize such emotions and the latter three are more subjective. While there is substantial research on global constructs of positive versus negative affect, Pekrun's model is unique since it specifically attends to different kinds of emotions pertaining to student behavior in real-world *academic* contexts. The two important dimensions in Pekrun's model of Achievement emotions are:

a) Valence, which describes positive (i.e., pleasant) states (e.g., enjoyment, happiness, etc.) that can be differentiated from negative (i.e., unpleasant) states (e.g., anger, anxiety, boredom, etc.)
b) Activation, wherein physiologically activating states can be distinguished from deactivating states, such as activating excitement versus deactivating relaxation.

At a higher level, it is important to distinguish between the degree of activation implied in emotional states even if they are under the same category of positive or negative. To illustrate, both anxiety and hopelessness are negative (unpleasant) emotions; however, their effects on students' learning outcomes can differ dramatically, as anxiety can motivate a student to invest effort in order to avoid failure, whereas hopelessness likely undermines any kind of engagement. At a lower level in each of the four categories, it is still necessary to distinguish between distinct emotions. For example, both anxiety and anger are activating negative emotions; however, whereas anxiety is associated with avoidance, anger is related to approach motivation (Carver & Harmon-Jones, 2009 as cited in Pekrun et al., 2012). Therefore, we will distinguish between Pekrun's 8-emotions from the Achievement emotions model[2], which will serve as the conceptual basis for our CNN algorithm.

Combining the valence and activation dimensions renders a 2 × 2 taxonomy of achievement emotions:

| | Positive | Negative |
|---|---|---|
| Activating | Enjoyment<br>Hope<br>Pride | Anger<br>Anxiety<br>Shame |
| Deactivating | Relief<br>Relaxation | Hopelessness<br>Boredom |

**Table 2.** Valence and activation dimensions of achievement emotions

However, we will also include a 9th emotion of neutrality to avoid the danger of missing important parts of students' affective experience while class is in session.

[2] **#levelsofanalysis:** We've conducted a multi-level analysis emotional states by identifying the categorical level (activating/deactivating, and positive/negative), and the elementary level (specific emotions in each category), with appropriate justification for each level. Later, we've also distinguished emotional intelligence as a salient feature in a classroom setting and measured the group-level phenomena using a validated questionnaire. This is combined with us establishing achievement emotions in an academic setting as an emergent property resulting from a student's internal state and classroom peer-to-peer and instructor-student interactions.

Connections between emotions and cognition have been explored. Positive activating task-related emotions (e.g. pride, enjoyment) promotes information retrieval (Kosslyn, 2017; Pekrun, Lichtenfeld, Marsh, Murayama, & Goetz, 2017), intrinsic motivation and elaborative learning strategies (Pekrun et al., 2011) (Figure 2), while the converse effect is true overall for negative emotions. The effects are currently ambiguous for positive deactivating emotions ("Achievement Emotions - Dr. Reinhard Pekrun," n.d.). The aforementioned arguments lead us to predict that there will be a positive association between positive, activating emotions experienced during an ALF class and individual academic performance (Hypothesis 1) and a negative association between negative emotions and individual academic performance (Hypothesis 2) for a Minerva sample.

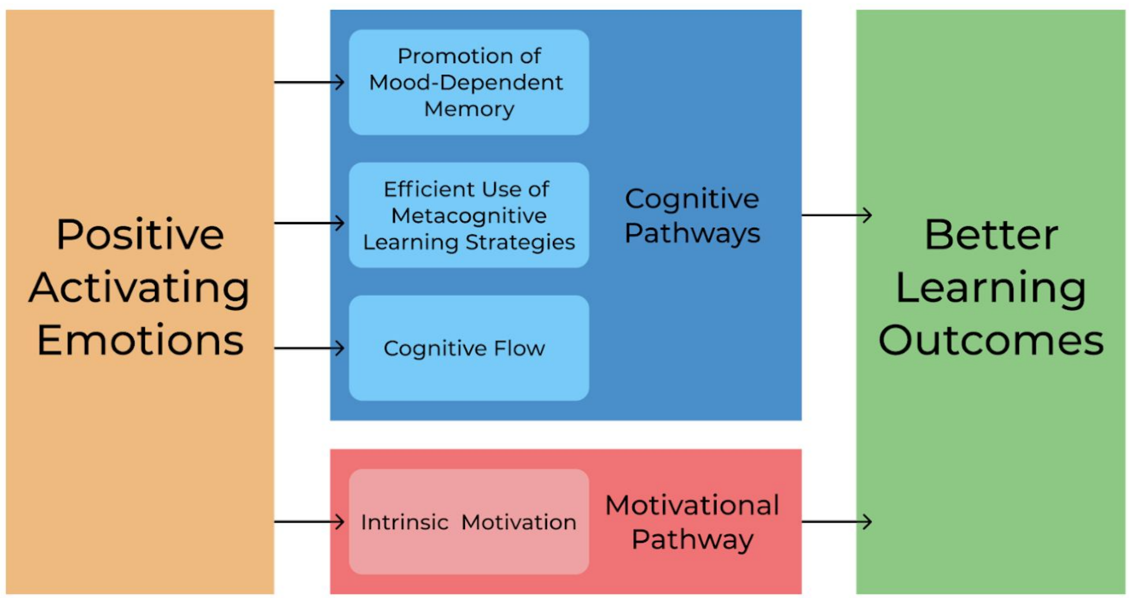



**Figure 1**. Conceptual map of the effects of positive, activating emotions on learning outcomes.

On a group level, the development of a positive, encouraging learning environment, i.e. with a high collective emotional intelligence (EQ), is central to students' educational experience (Augustsson 2010; Vandervoort 2006). There is a significant relationship between the total emotional intelligence in a classroom and individual learning strategies (Hasanzadeh & Shahmohamadi, 2011). A positive affective environment is conducive to higher levels of intrinsic motivation, more cognitive activity during lessons and positive attitude towards schools and academics (Aritzeta et al., 2016) which, in turn, predict achievement (Ruthig et al., 2008). Conversely, the experience of rejection and isolation is associated with behavioural problems in the classroom and lower achievement (Osterman, 2000). Thus, we hypothesize that in classes where, on average, students achieved high academic grades, perceived group EI operationalized with Aritzeta et al.'s questionnaire, will be higher (Hypothesis 3).

Emotions cannot be directly measured because they are conceptual entities (constructs). However, they emerge from environment-person interactions (context) and influence action by modulating cognition. Therefore, it should be possible to "infer" emotion by analyzing the unfolding context and learner actions. For instance, emotions activate bodily response systems for actions and make it possible to infer learner affect (a latent variable) from machine-readable bodily signals (observables). We will explore the avenue of automatic detection of emotion from facial features and body movements.

As an example, consider Pardos et al. (2013) which developed emotion detectors for ASSISTments, a web-based intelligent tutoring system which applied common ML classification algorithms but performance was limited to an accuracy of .632 (Pardos et al., 2013).

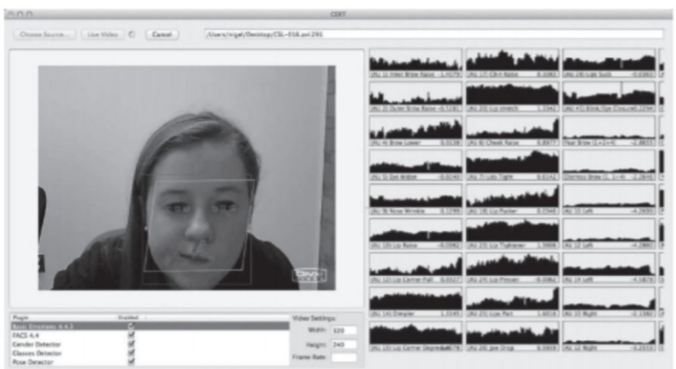


**Figure 2**. Automatic tracking of facial features using the Computer Expression Recognition Toolbox (CERT).

As a second example, consider the Computer Expression Recognition Toolbox (CERT) (Figure 3), which measures facial expressions in real-time (Bartlett, Littlewort, Wu, & Movellan, 2008). In various experiments, CERT was able to differentiate real from fake pain (Littlewort, Bartlett, and Lee, 2009), detect driver drowsiness above 98% accuracy (Vural et al., 2007), predict perceived difficulty of a video lecture and preferred presentation speed (Whitehill et al., 2008). CERT is a powerful method for extracting facial expressions, since it is based on the Facial Action Coding System (FACS) of Ekman et al (2005). However, CERT was based on a heuristic-coding scheme and is thus insensitive to nuances of intensity and variation in a facial expression.

For professors, understanding where students are struggling can help offer more personalized responses. Curriculum designers assess detected emotion in relation to specific segments of content and provide appropriate supporters to learners. Cornerstone professors (McAllister, 2018; Bogucki, 2018; Chun, 2018) reported that the problem of real-time emotion recognition is compelling and data on emotional states can help them adjust teaching strategies. Although the emotional states of instructors are known to influence student affect and engagement (Keller et al., 2014), teachers report suppressing their emotions, emphasizing the importance of neutral expression while in class. Therefore, instructor emotion recognition is excluded from the scope of our proposal.[3]

With all the importance of emotion recognition in educational settings emphasized, in this paper, we propose a system that can identify and monitor emotions of the student in the ALF and provide a real-time feedback mechanism for instructors to promote better learning outcomes.

## *2. Proposed Technical Approach*

### 2.1 Convolutional Neural Network (CNN) Architecture Design

CNNs have been proven to be the most promising in image classification and emotion recognition in video stream input (Ouellet, S., 2014), with image classification accuracy being as high as 0.96 (He, K. et al., 2016). Therefore, we will use CNNs to classify emotional expressions demonstrated in static images. To detect emotions in live video stream, we use the same method exploited in Duncan et al. – attaching the video stream to a face detector that crops one image per 150 ms, and feeding the extracted images of faces to the CNN that outputs the result for a forward pass in 200 ms.

In this chapter, the functioning of CNNs in the context of emotion recognition will be introduced with detailed descriptions of the algorithmic steps and structures used. Because CNNs have the basis of regular neural networks (RNNs), and because the only difference between CNNs and RNNs is architecture choices, as the algorithms involved remain the same, an overview of RNNs will also be discussed.

[3] **#rightproblem:** throughout this literature review and hypothesis testing, we are making a good case for why in-class emotion recognition is important for Minerva students' learning; the goal state is to have a system that can register student's experienced emotional states in class for better self-awareness and constructive feedback and modified teaching strategies from professors, compared to the current state that lacks such systems; the scope has been clearly defined (e.g., restricted to only achievement emotions, and not include professors' emotions in our solutions); obstacles (the difficulty of correctly classifying students' emotions due to excessive movements, subtle facial expressions, etc.) are also specified in the discussion of ASSISTments and CERT.

**A. Regular neural network**

a. Forward pass

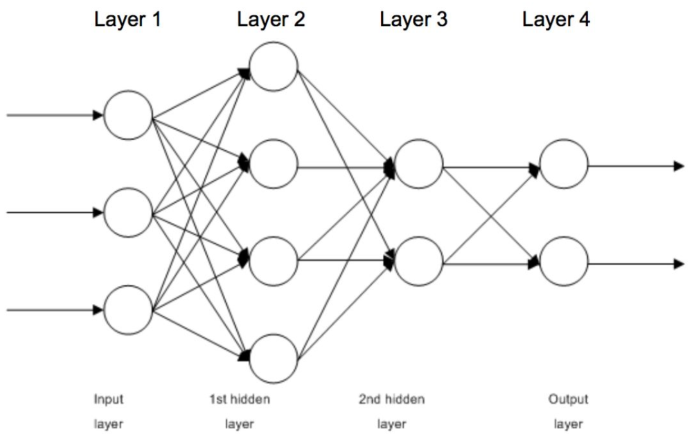



**Figure 7.** Visualization for a regular neural network that has three layers (two hidden and one output layers)

In general, a neural network includes an input layer, hidden layers, and an output layer. In Figure 1, the input layer has three neurons while the output layer has two.

Specifically, the net receives inputs from one image, which are numbers representing pixels of the image. We will feed the network with images that have width of 32, height 32, and three color channels Red, Green, and Blue (RGB), which makes for $32 \times 32 \times 3 = 3072$ neurons in the input layer. Since we want to classify a facial expression shown in the image into one of the nine classes (enjoyment, hope, pride, anger, anxiety, shame, relief, relaxation, neutral, hopelessness, boredom), the output layer will have nine neurons, each holding a *score:* a higher score of an output neuron means a higher probability that the expression belongs to that class.

Some notations:[4]

[4] **#audience:** different materials in the field denote and call the concepts or components of neural networks differently. By specifying the notations here and, in the text above, introducing and defining some key concepts (neurons, input layer, scores, activation, etc.), we do not want to assume the readers' familiarity with our own notation scheme and word choice and therefore make sure they can follow the essay easily. The table also allows for quick reference for the readers in case they forget which sign denotes which concept.

N: number of training images
$a_i^l$: the score (the *activation)* of the $i$-th neuron in the $l$-th layer
$s_i$: the activation of the $i$-th neuron in the output layer
$w_{ij}^l$ $(l \neq 1)$: the weight for the connection between the j-th neuron in the $(l-1)$-th layer and the $i$-th neuron in the $l$-th layer
$b_i^l$: bias term for the $i$-th neuron in the $l$-th layer.
$n^l$: number of neurons in the $l$-th layer
$z_i^l$: intermediate quantity for the $i$-th neuron in the $l$-th layer
$g$: activation function
$\lambda$: regularization strength
$\alpha$: learning rate
$k$: number of iterations

Except for the input layer, where the activations depend on the input images, all the activations in later layers are computed as follows:

$$z_i^l = \sum_{j=1}^{n^{l-1}} w_{ij}^l a_j^{l-1} + b_i$$

$$a_i^l = g(z_i^l)$$

Combining the two formulae, we get:

$$a_i^l = g(\sum_{j=1}^{n^{l-1}} w_{ij}^l a_j^{l-1} + b_i) \qquad \textbf{(1)}$$

There are many options for the function $g$, but we will apply the ReLU (Rectified Linear Unit) function for the hidden layers:

$$g(x) = max(0, x)$$

and return the output layer without any activation:

$$g(x) = x$$

Compared to other possible activation functions (e.g., sigmoid, tanh, maxout), ReLU is empirically shown to be superior in enhancing the network's performance (Krizhevsky, A. et al., 2012)

b. Gradient descent algorithm
    i. The loss function

The scores in the output layer are then logged into the Softmax cross-entropy loss function L, which has nice probabilistic interpretations and has been used commonly; for example, in Eigen, D. et al., 2015; Ronneberger, O. et al., 2015 ):

$$L_i = -log\frac{e^{s_{y_i}}}{\sum_j e^{s_j}} \qquad (2)$$

$$L = \frac{1}{N}\sum_{i=1}^{N} L_i + \frac{\lambda}{2}\sum_{l:l\neq 1}\sum_{m,n}(w_{mn}^l)^2 \qquad (3)$$

Where $s_{y_i}$ is the score for the correct emotions expressed in image $i$. The intuition behind formula (2) is, when $s_{y_i}$ is small, which we do not want, $L_i$ is large, so the function penalizes incorrect classification and our goal is to minimize the loss L. The latter part in (3) starting with $\frac{\lambda}{2}$ is the regularization term, which helps preventing the network from overfitting (Scholkopf, B. et al., 2001).

The job of minimization can be efficiently facilitated by gradient descent.

ii. Gradient descent[5]

Equation (3) requires computing the loss for *all* training images. This task is too computationally and spatially expensive, given the huge number of input features and parameters. To combat this, we will use *stochastic gradient descent* (SGD) (Bottou, L., 2010), in which only one image is randomly selected for loss calculation at each iteration. Therefore, the loss at each iteration now becomes:

$$L = -\log\frac{e^{s_{y_i}}}{\sum_j e^{s_j}} + \frac{\lambda}{2}\sum_{l:l\neq 1}\sum_{m,n}\left(w_{mn}^l\right)^2 \qquad (4)$$

However, since SGD deals with heavy randomness in each iterations, it introduces more noise and sometimes forestalls the model from converging to a minima due to fluctuations in weights. Since the fluctuations around the local minimum is proportional to the learning rate $\alpha$ (LeCun, Y. et al., 1998), we will reduce $\alpha$ after a certain number of iterations $k_0$ (i.e., $k_0$ iterations make up an *epoch*, after which $\alpha$ will be reduced)

The gradient descent algorithm is as follows:

**Gradient Descent Algorithm**

**Step 1.** Set the total number of iterations $k$

**Step 2.** Set the number of iterations per epoch $k_0$ (to decrease $\alpha$)

**Step 3.** Set the decreasing rate for the learning rate $\gamma$ (another hyperparameter)

**Step 4.** Create a variable $t$ to track the number of iterations

**Step 5.** Assign $t$ to 1

**Step 6.** Choose a random value of $n$ ($n \in [1;N]$)

**Step 7**. Implementing the forward pass for the $n$-th training image $x_n$ with the correct class $y_n$

**Step 8.** Compute the loss L

**Step 9.** Compute the gradients $\frac{\partial L}{\partial w_{ij}^l}$ and $\frac{\partial L}{\partial b_i^l}$ for all possible $(l,i,j)$ $(l \neq 1)$

**Step 10.** Update the weights: $w_{ij}^l := w_{ij}^l - \alpha\frac{\partial L}{\partial w_{ij}^l}$

**Step 11.** Update the biases: $b_i^l := b_i^l - \alpha\frac{\partial L}{\partial b_i^l}$

**Step 12.** Increment $t$ by 1

**Step 13.** If $t \% k_0 = 0$ (which means an epoch is over), update $\alpha$: $\alpha := \gamma\alpha$; otherwise jump to step 14.

**Step 14.** If $t \leq k$, jump to step 4, otherwise jump to step 15

**Step 15.** Exit

Below is our snippet code of SGD for a two-layer RNN:

```
def gradient_descent(X, y, learning_rate=1e-3,
                     learning_rate_decay=0.95,
                     reg_strength=5e-6, num_iters=1000,
                     batch_size=1, iterations_per_epoch=100)
                     # in SGD, batch_size equals 1, since we only randomly
                     # choose 1 training example per iteration.

  for i in range(num_iters):

    index = np.random.choice(np.arange(N), batch_size) #choose a random number n from 0 to N-1
    X_batch = X[index] # pick the randomly chosen image
    y_batch = y[index]

    loss, grads = loss(X_batch, y=y_batch, reg_strength=reg_strength) #calculate the loss and grandient

    #update the weights and the biases

    W2 -= learning_rate * grads['W2']
    W3 -= learning_rate * grads['W3']
    b2 -= learning_rate * grads['b2']
    b3 -= learning_rate * grads['b3']

    #update the learning rate after 100 iterations
    if (i+1) % iterations_per_epoch == 0:
        learning_rate *= learning_rate_decay

  return (W2,W3,b2,b3)
```

The algorithm does not guarantee a convergence to a minima. However, the overarching idea of gradient descent is since the quantity $\frac{\partial L}{\partial w_{ij}^l}$ depicts the rate of an *increase* of L if there is an increase in $w_{ij}^l$ (i.e., if $w_{ij}^l$ increases by $h$, L will increase by $h\frac{\partial L}{\partial w_{ij}^l}$). Therefore, when updating weights by *decreasing* them by a multiplication of their gradients, we are essentially moving L towards one of its minimum, given a small enough $\alpha$. However, applying SGD renders more flexibility and sometimes L is likely to diverge away from its minima in an iteration, but overtime, it will show convergence at a rate faster than batch-learning (LeCun, Y. et al., 1998).

c. Back propagation (backprop) algorithm[6]

**Step 9** in the SGD is no easy task, especially when the CNNs involve multiple layers. The back propagation is an algorithm to make the task of computing gradients more convenient.
First, we introduce two quantities, $L^{(1)}$ and $L^{(2)}$:

$$\text{If } L^{(1)} = -log\frac{e^{s_{y_n}}}{\sum_j e^{s_j}} \text{ and } L^{(2)} = \frac{\lambda}{2}\sum_{l:l\neq 1}\sum_{m,n}(w_{mn}^l)^2$$

$$\text{then } L = L^{(1)} + L^{(2)} \text{ and } \frac{\partial L}{\partial w_{ij}^l} = \frac{\partial L^{(1)}}{\partial w_{ij}^l} + \frac{\partial L^{(2)}}{\partial w_{ij}^l};\ \frac{\partial L}{\partial b_i^l} = \frac{\partial L^{(1)}}{\partial b_i^l} + \frac{\partial L^{(2)}}{\partial b_i^l} = \frac{\partial L^{(1)}}{\partial b_i^l}$$

[6] **#algorithm:** show a detailed step-by-step algorithm and how the algorithm can work by proving the involved equations and their proofs

We also define another quantity, the *error of a neuron,* $\delta$. The error of the $i$-th neuron in the $l$-th layer is:

$$\delta_i^l = \frac{\partial L^{(1)}}{\partial z_i^l}$$

The backprop algorithm in an iteration of gradient descent is as follows:

**Step 1**. Compute the errors of all the neurons in the output layer ($l^*$ represents the last/output layer):

$$\delta_i^{l^*} = \frac{\partial L^{(1)}}{\partial a_i^{l^*}} g'(z_i^{l^*}) \qquad \textbf{(a)}$$

which gives us:

$$\delta_i^{l^*} = \begin{cases} \frac{e^{s_i}}{\sum_j e^{s_j}}, & \text{if } i \neq y_n \\ \frac{e^{s_i}}{\sum_j e^{s_j}} - 1, & \text{if } i = y_n \end{cases}$$

**Step 2.** Calculate the gradients of all the biases in the output layer:

$$\frac{\partial L}{\partial b_i^{l^*}} = \frac{\partial L^{(1)}}{\partial b_i^{l^*}} = \delta_i^{l^*} \qquad \textbf{(b)}$$

**Step 3.** Calculate the gradients of all the weights of the output layer:

$$\frac{\partial L^{(1)}}{\partial w_{ik}^{l^*}} = a_k^{l^*-1} \delta_i^{l^*} \qquad \textbf{(c)}$$

which gives us the gradient:

$$\frac{\partial L}{\partial w_{ik}^{l^*}} = \frac{\partial L^{(1)}}{\partial w_{ik}^{l^*}} + \frac{\partial L^{(2)}}{\partial w_{ik}^{l^*}} = a_k^{l^*-1} \delta_i^{l^*} + \lambda w_{ik}^{l^*}$$

**Step 4.** Create variable $l$ to track the layer we are working on

**Step 5.** Assign $l^* - 1$ to $l$ (we have worked with the $l^*$-th layer in the above steps; we will keep working backward till $l = 2$)

**Step 6.** Compute the errors of all the neurons in the hidden layer $l$:

$$\delta_i^l = \sum_j w_{ji}^{l+1} \delta_j^{l+1} g'(z_i^l) \qquad \textbf{(d)}$$

where

$$g'(z_i^l) = \begin{cases} 1, & \text{if } z_i^l > 0 \\ 0, & \text{if } z_i^l < 0 \end{cases}, \text{ since } g(x) = max(0, x), \text{ and } g(x) \text{ is not differentiable at } 0$$

**Step 7.** Calculate the gradients of all the biases in the hidden layer $l$:

$$\frac{\partial L}{\partial b_i^l} = \frac{\partial L^{(1)}}{\partial b_i^l} = \delta_i^l \qquad \textbf{(e)}$$

**Step 8.** Calculate the gradients of all the weights of the hidden layer $l$:

$$\frac{\partial L^{(1)}}{\partial w_{ik}^l} = a_k^{l-1} \delta_i^l \qquad \textbf{(f)}$$

which gives us the gradient:

$$\frac{\partial L}{\partial w_{ik}^l} = \frac{\partial L^{(1)}}{\partial w_{ik}^l} + \frac{\partial L^{(2)}}{\partial w_{ik}^l} = a_k^{l-1} \delta_i^l + \lambda w_{ik}^l$$

**Step 9.** Decrease $l$ by 1

**Step 10.** If $l$>1, jump to step 6; otherwise jump to step 11

**Step 11.** Exit

To show how the algorithm works, we just need to prove the five fundamental equations **(a)-(f)**. Proofs for **(a)** and **(d)** are provided by Nielsen, M. A. (2015). Since **(e)** and **(f)** are generalizations of **(b)** and **(c)**, respectively, we will only prove (e) and (f):[7]

**Proof for (e):**

Applying the chain rule, we have:

$$\frac{\partial L^{(1)}}{\partial b_i^l} = \sum_j \frac{\partial L^{(1)}}{\partial z_j^l} \frac{\partial z_j^l}{\partial b_i^l}$$

Since $z_j^l$ is independent of $b_i^l$ if $j \neq i$ ( $\frac{\partial z_j^l}{\partial b_i^l} = 0$ if $j \neq i$ ) :

$$\frac{\partial L^{(1)}}{\partial b_i^l} = \frac{\partial L^{(1)}}{\partial z_i^l} \frac{\partial z_i^l}{\partial b_i^l} = \frac{\partial L^{(1)}}{\partial z_i^l}.1 = \frac{\partial L^{(1)}}{\partial z_i^l} \ (*)$$

Moreover by definition we have:

$$\delta_i^l = \frac{\partial L^{(1)}}{\partial z_i^l} \ (**)$$

Therefore,

$$\frac{\partial L^{(1)}}{\partial z_i^l} = \delta_i^l \text{ (deducted from (*) and (**))}$$

**Proof for (f):**

Applying the chain rule, we have:

$$\frac{\partial L^{(1)}}{\partial w_{ik}^l} = \sum_j \frac{\partial L^{(1)}}{\partial z_j^l} \frac{\partial z_j^l}{\partial w_{ik}^l}$$

---

[7] **#deduction:** the proofs for (e) and (f) implicitly invoke multiple, interwoven deductive rules. For example, for the proof for (e) (we want to prove that A=B):

| | |
|---|---|
| 1. If (A=C and B=C) then (A=B) | premise |
| 2. A=C | premise |
| 3. B=C | premise |
| 4. A=C and B=C | adjunction, 2, 3 |
| 5. A=B | modus ponens, 1, 4 |

Since $z_j^l$ is independent of $w_{ik}^l$ if $j \neq i$ ( $\dfrac{\partial z_j^l}{\partial w_{ik}^l} = 0$ if $j \neq i$ ) :

$$\frac{\partial L^{(1)}}{\partial w_{ik}^l} = \frac{\partial L^{(1)}}{\partial z_i^l}\frac{\partial z_i^l}{\partial w_{ik}^l} = \delta_i^l \times a_k^{l-1}$$

d. Neural network synthesis[8]

We now have the entire algorithm to train a RNN:

**Step 1.** Initialize the weights: drawing from a Gaussian distribution with mean 0 and small standard deviation $\sigma$ (e.g., 1e-10)

**Step 2.** Initialize the biases: set all to 0

**Step 3.** Set the total number of iterations $k$

**Step 4.** Set the number of iterations per epoch $k_0$ (to decrease $\alpha$ )

**Step 5.** Set the decreasing rate for the learning rate $\gamma$ .

**Step 6.** Create a variable $t$ to track the number of iterations

**Step 7.** Assign $t$ to 1

**Step 8.** Choose a random value of $n$ ($n \in [1;N]$)

**Step 9**. Implementing the forward pass for the $n$-th training image $x_n$ with the correct class $y_n$

**In this step, calculate $z_i^l$ and $a_i^l$ based on the current weights and biases**

[8] **#breakitdown:** the job of training a neural network requires many layers of algorithms and would seem daunting at first if taken as a whole. We appreciated the approach that breaks this task down into three components: the forward pass, the gradient descent, and the backprop and later showed how the components can be combined to a full working solution.

**Step 10.** Compute the loss L

**Step 11.** Compute the gradients $\frac{\partial L}{\partial w^l_{ij}}$ and $\frac{\partial L}{\partial b^l_i}$ for all possible $(l,i,j)$ $(l \neq 1)$ :

**In this step, implement the 11 steps of backprop algorithm**

**Step 12.** Update the weights: $w^l_{ij} := w^l_{ij} - \alpha \frac{\partial L}{\partial w^l_{ij}}$

**Step 13.** Update the biases: $b^l_i := b^l_i - \alpha \frac{\partial L}{\partial b^l_i}$

**Step 14.** Increment $t$ by 1

**Step 15.** If $t \% k_0 = 0$ (which means an epoch is over), update $\alpha$ : $\alpha := \gamma\alpha$ ; otherwise

jump to step 14.

**Step 16.** Exit (training accomplished)

## B. Convolutional neural network

### a. Differences

CNNs are a refined idea of RNNs. While the algorithmic steps are the same, unlike RNNs, which stretches out the features of the images into a column vector, CNNs preserve the spatial structure of the images' features. In other words, if an image has size $a \times b \times c$, the inputs fed into CNNs are still a three-dimensional vector of the same size. The outputs at each layer, thus, are also 3D volumes. Some basic types of layers in CNNs include:[9]

1. Input layer: holds the raw pixel values of the images preserved in 3D volumes.
2. Convolutional layer: computes the output of the neurons like in RNNs, but instead of connecting with all the neurons in the previous layer with a different weight for each neuron, it has a much smaller number of weights that are connected to only a small region in the previous layer. This small set of weights is called a *filter.* The convolutional layer will slide this filter from one region to another to produce the outputs until it covers all the neurons in the previous layer. Thus, neurons of different regions share the same weights This significantly reduces the number of weights the network has to train and, therefore, one can have a deeper network with a smaller number of parameters. This makes CNNs superior to RNNs. Deeper layers have been shown to facilitate high performance of a CNN:

---

[9] **#emergentproperties:** Given the various complicating, randomness-inducing elements in the CNNs (e.g., the randomized initialization of hyperparameters, the randomization in choosing the input in SGD), which lead to unpredictability of the CNN as for exactly how long it takes to converge, how well it will converge, etc., we can treat the CNN itself as a complex system. The emergent property here is the CNN's ability to classify the emotions at the output layer as a results of the interactions of multiple intermediate layers of different types.

| Network | Number of layers | Testing accuracy |
|---|---|---|
| AlexNet<br>(Krizhevsky, A. et al., 2012) | 8 | 83.6% |
| VGGNet<br>(Simonyan, K. et al., 2014) | 19 | 92.7% |
| GoogleNet<br>(Szegedy, C. et al., 2015) | 22 | 93.3% |
| ResNet<br>(He, K. et al., 2016) | 152 | 96.43% |

**Figure 8.** Number of layers and testing accuracy of different Convolutional Neural Networks. The four networks are winners of ImageNet Large Scale Visual Recognition Challenge. Testing data are data from ImageNet.

3. ReLU layer: subjects the outputs of the previous layer to ReLu function
4. Pooling layer: downsizes the outputs of the previous layer
5. Fully-connected layer: like layers in RNNs, this layer connects to all of the activations in the previous layer

b. Implementation:

We will collect our own data to train the CNN. This includes images of Minerva students' faces extracted from class videos. Due to time and money constraints, we may not have a large data set, which could easily lead to an overfitting model. We will adopt the transfer learning method used in Duncan et al.. Transfer learning is a technique of utilizing a model trained from a previous dataset and applying to another task (Goodfellow, I. et al., 2016). In this case, we will reuse the VGG_S model by Levi, G. et al. (2015), which was trained on multiple huge datasets like CASIA WebFace (Yi, D. et al., 2014) and SPEW (Dhall, A. et al., 2011), and apply transfer learning. As aforementioned, we expect our own dataset of Minerva students' faces to be small. Therefore, it is not ideal to retrain the entire VGG_S CNN, because it would likely lead to overfitting. Moreover, because we also expect our own dataset to be similar to the datasets originally used to train VGG_S (since they are all face-based datasets and are trained for the task of emotion detection), we will retrain the last fully-connected layers in the VGG_S. Multiple research have shown that in early layers, CNNs only learn the generic features; it is only at the last layers does the CNN learn the most data-specific features (Larochelle, H. et al., 2009; Hinton, G. E. et al., 2006). Therefore, it is reasonable to only fine tune the last layers of VGG_S to adapt it to our Minerva dataset.

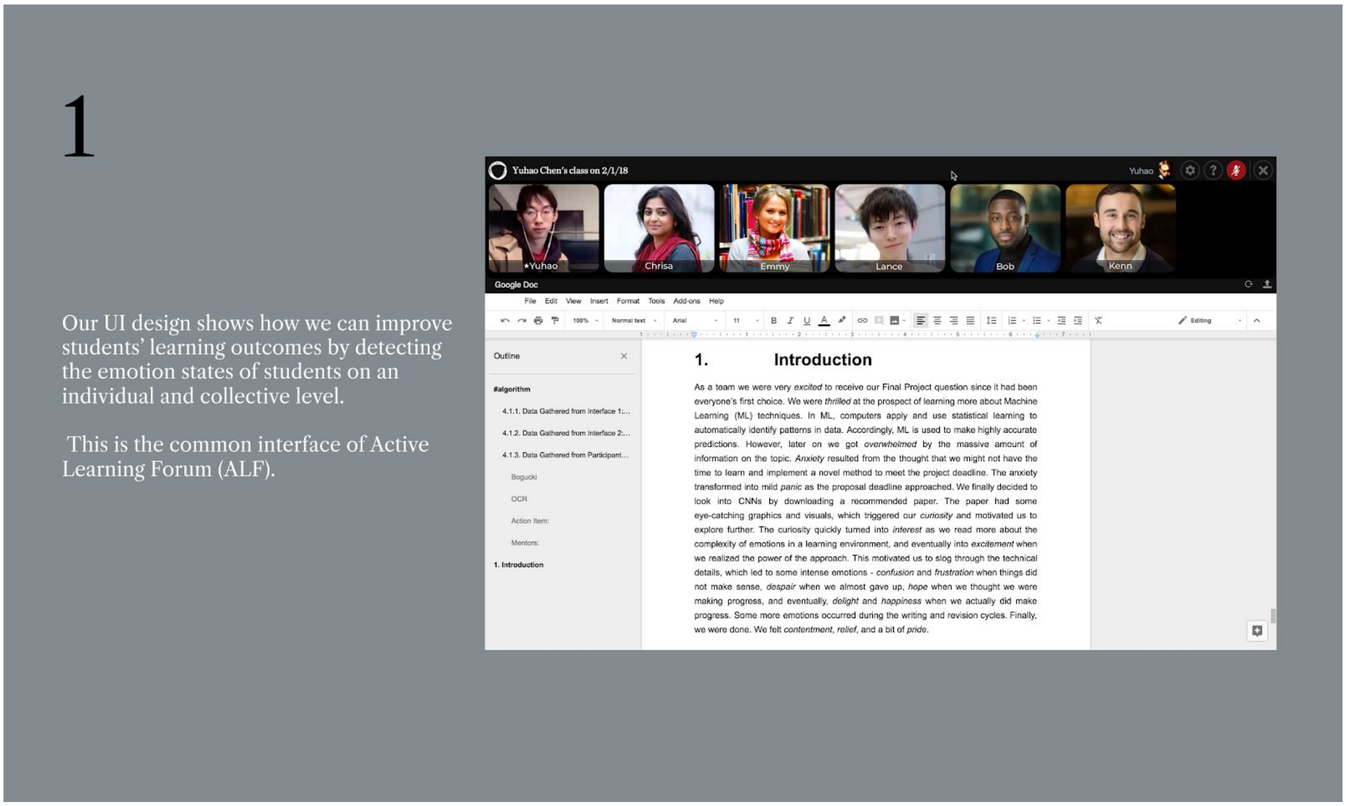
This is an application of discerning an analogy with transfer learning in human, which is the process of recalling and applying knowledge in new situations (Salomon, G. et al., 1987). In this case the VGG_S is used to apply what it has learned to a completely new dataset of Minerva students. Also, transfer learning in humans is characterized by the act of forming a bridge between knowledge and practice (Zimrot, R. et al., 2007). In the context of CNN transfer learning, the VGG_S is treated as an entity carrying with it the “knowledge” gained from training on the original data, and now our job is to put the previously acquired knowledge into practice, which is classifying emotions of Minerva students. However, there are some systematic dissimilarities since the idea of neural networks is to put the data into a black box which contains multiple layers and wait for the output, while learning in human discourages such black boxes (i.e., students are encouraged to know the underlying mechanisms, reasons, etc. (Kosslyn, S. M. (in press)). Therefore, we cannot apply the techniques outlined in Kosslyn (in press) to CNNs. However, in human transfer learning, both similarities and differences in old and new contexts are important for effective learning (Thorndike, E. L., 1901;  Marton, F., 2006) and transfer learning in CNNs applies just this phenomena: they strive to maintain high-level similar features in early layers and modify the last layers to allow the network to pick up the differences specific to the current data. **[10]**

However, the Minerva dataset may pose a significant hardship to the CNN when it is used as a real-world application due to large variances in face pose and illumination. All these may significantly reduce the network's accuracy. Future works of modifying the current preliminary CNN may include complementing the network with more multimodal features (Kaya et al., 2017)

[10] **#analogy:** point out not only the similarities but also the differences between the problem of effective transfer learning in human and in machine to see what solutions are applicable; the solution for effective transfer learning in CNNs is detailed in the previous paragraph

**2.2 User Interface Design**[11, 12]

We will analyze students' facial expressions by processing the videos using the algorithms mentioned above and store the data acquired to transfer to visible notifications and graphs for users. This part presents what the UI will look like.



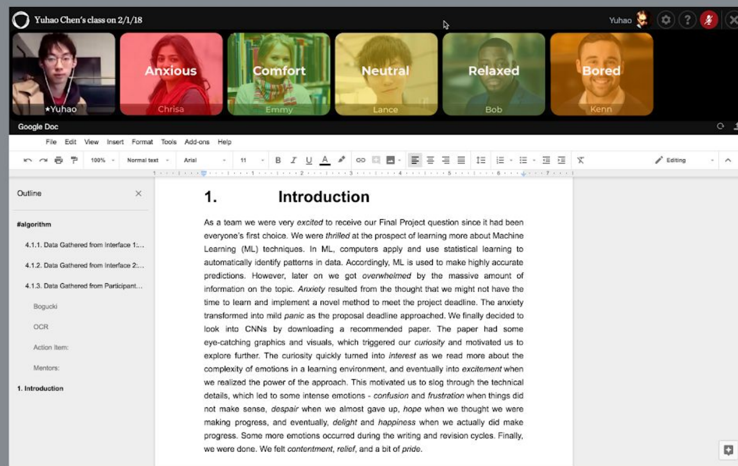

[11] **#communicationdesign:** We applied principles of discriminability and compatibility outlined by Kosslyn et al. (2012). For discriminability, in the third slide, we make the word that indicates the collective class emotion larger than other elements and usde the white typeface distinct from the dark background color to make it salient and emphasize its important. For compatibility, for example, in the fourth slide, we chose the red color, which is often associated with danger, to make it compatible with the message convey ("negative class emotion") to warn the professor.

[12] **#designthinking:** We conducted intensive user research using appropriate sampling methods to understand what capabilities and features our product needs to provide. We generated many different sketches for each feature and sought peers and instructors' feedback to iterate over our product design and make it as intuitive and self-explanatory as possible through appropriate **#cognitivepersuasion** techniques.

# 2

In this step, the instructor can see everyone's current emotional state by pressing the F key. Red implies negative emotions, green implies positive emotions, and yellow implies a neutral state.

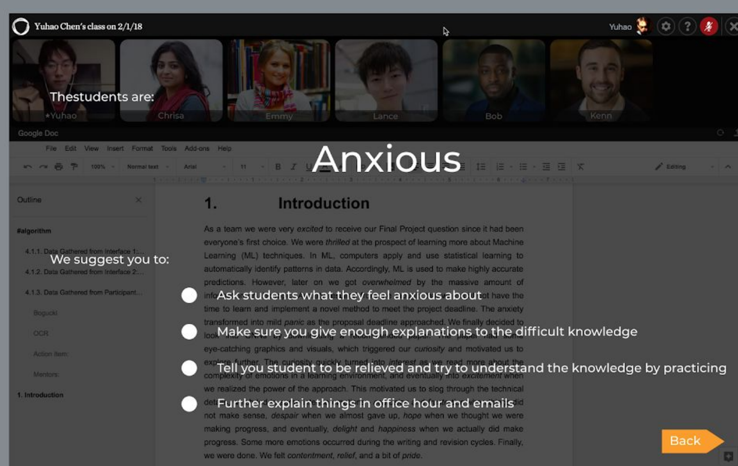
The specific emotion experienced by the student will appear on top of the color filter. Instructors have the flexibility to adjust their teaching strategies according to the class' emotional state.




# 3

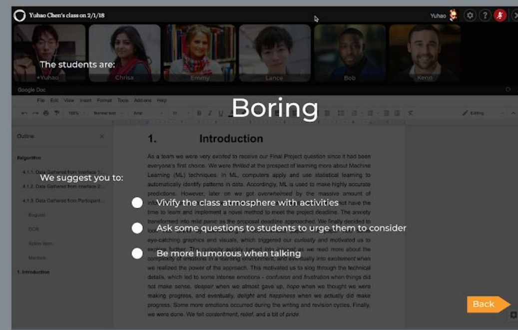
The instructor can see the collective emotional state for the whole class by pressing ctrl+F key. The interface also displays recommendations accordingly.

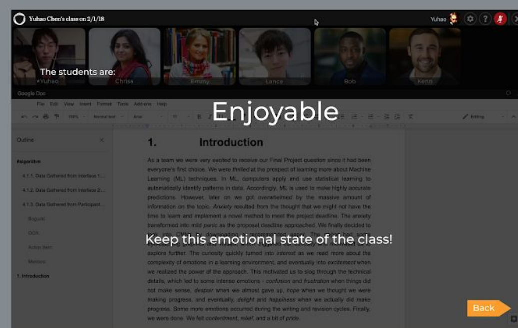
For example, in this page, after analyzing students' facial expression over the past 2 minutes, the system generates the result that the collective emotion is Anxious, and makes appropriate suggestions.



These are additional pages showing two other possible emotional states in the class.

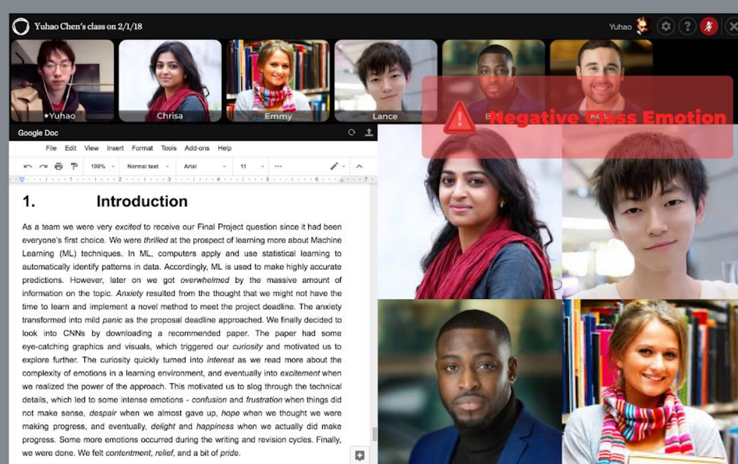
As previously stated, in addition to idenitifying the emotional state, our software product will automatically show a series of suggestions to help the teachers to promote a positive class atmosphere.



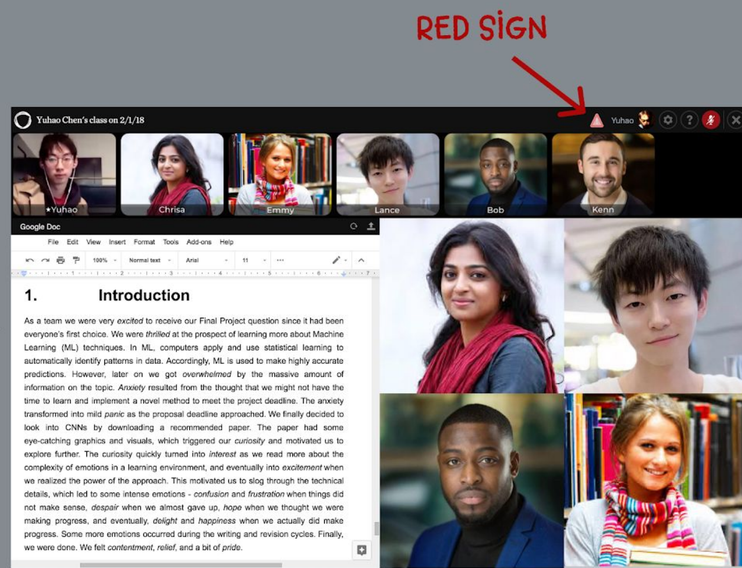




# 4

In this step, the red box notifying the instructor of ‘Negative Class Emotion’ pops up on the teachers’ screens for 3 seconds whenever more than half of the students in the class experience negative emotions.



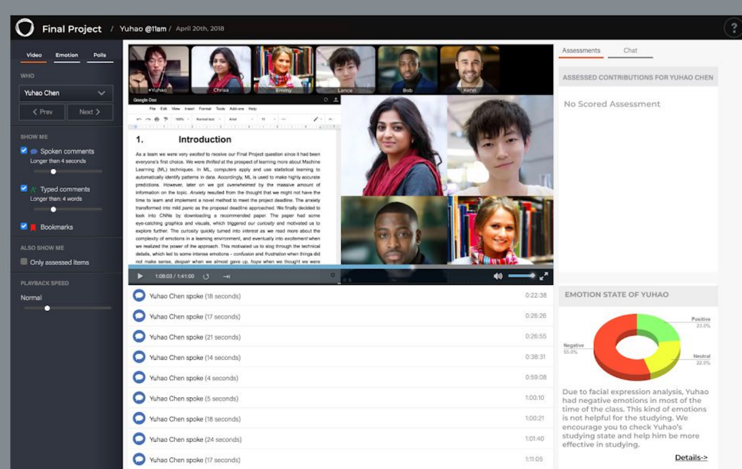
After 3 seconds, a small red sign on the top of the screen will replace the negative emotion notification. The sign will disappear only when less than 1/3 of the students in the class have a positive emotion.




# 5

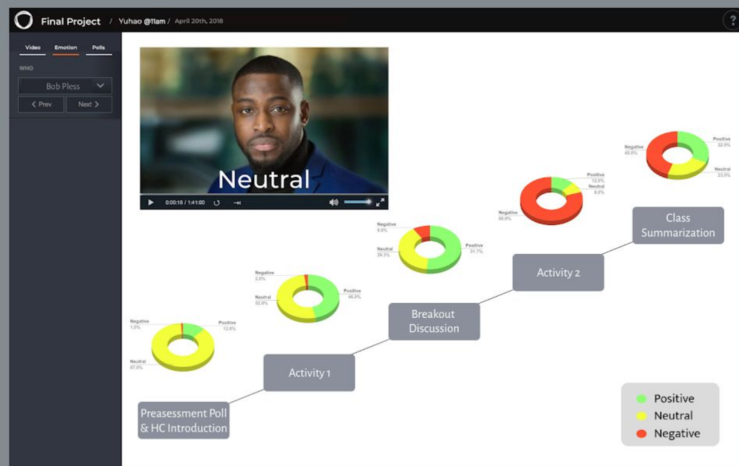
This new interface for reviewing class recordings displays an analysis of the emotional state of each student at the right bottom of the page.

Students and instructors can find the proportion of positive/negative/neutral emotions in this class, thereby better assessing their own performance and efficiency of the class.



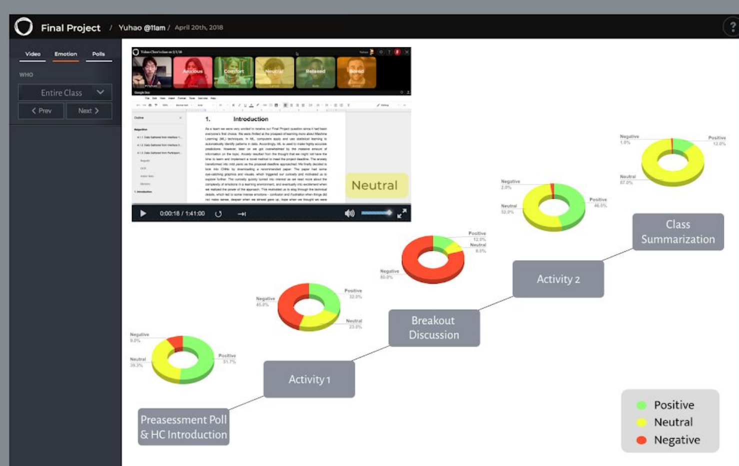
By pressing 'details' button on the last interface or click on the 'emotion' category in the left top, users can enter this interface. The Emotion tab contains the emotional trajectory for each individual student. Summary statistics are presented for each section of the class, along with the student's videostream.

The design is clean and straightforward, no instruction is needed.




The Emotion tab also displays summary of emotional trajectories for the entire session, divided according to in-class activities.

The process is designed to help teachers understand the overall affective state of the class, identify which emotions were dominant during specific activities and test whether any changes in instructional strategy led to positive academic emotions.



### *3. Future Plans*

Our CNN technology can be used to automatically analyse teachers' instructional practices, by extracting information on what teachers are doing (e.g. detect teacher questions) and contextualize how students are feeling, which in turn influences what they think, do and learn. Moreover, changes in students emotions can be analyzed across time. There are many questions that can be asked along this front. How do emotions change over the duration of an activity, a session, or the entire course? What is the affective trajectory of top performing students? What are the key emotional transitions that are indicative of learning? Further research is needed to map emotion trajectories over the duration of the course so that we better understand the relationships between emotions, their temporal dynamics, and educational outcomes.

*Word Count: 3655*

### *4. Bibliography*[13]

Achievement Emotions - Dr. Reinhard Pekrun. (n.d.). Retrieved April 18, 2018, from https://mediaspace.msu.edu/media/Achievement+Emotions+-+Dr.+Reinhard+Pekrun/1_vvf2z66r

Alonso, R., Brocas, I., & Carrillo, J. D. (2014). Resource Allocation in the Brain. *The Review of Economic Studies*, *81*(2), 501–534. https://doi.org/10.1093/restud/rdt043

An Evolutionary Approach to Understanding Distinct Emotions - Jessica L. Tracy, 2014. (n.d.). Retrieved April 18, 2018, from http://journals.sagepub.com/doi/abs/10.1177/1754073914534478

Aritzeta, A., Balluerka, N., Gorostiaga, A., Alonso-Arbiol, I., Haranburu, M., & Gartzia, L. (2016). Classroom emotional intelligence and its relationship with school performance. *European Journal of Education and Psychology*, *9*(1), 1–8. https://doi.org/10.1016/j.ejeps.2015.11.001

Barth, C. M., & Funke, J. (2010). Negative affective environments improve complex solving performance. *Cognition and Emotion*, *24*(7), 1259–1268. https://doi.org/10.1080/02699930903223766

Bartlett, M., Littlewort, G., Wu, T., & Movellan, J. (2008). Computer Expression Recognition Toolbox. In *2008 8th IEEE International Conference on Automatic Face Gesture Recognition* (pp. 1–2). https://doi.org/10.1109/AFGR.2008.4813406

[13] **#sourcequality:** We reviewed and referred to numerous high-quality, peer-reviewed, and, when applicable, most recent academic sources to obtain updated and reliable information. We also sought information from all relevant stakeholders e.g. from students through rigorous surveying methods and professors through qualitative interviews.

Berger, G., & Memisevic, R. (2016). Incorporating long-range consistency in CNN-based texture generation. *arXiv preprint arXiv:1606.01286*.

Bogucki, M. (2018, March). Personal interview.

Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010* (pp. 177-186).

Physica-Verlag HD.

Brown AL, Bransford JD, Ferrara R, Campione J. (1983). Learning, remembering, and understanding. In: Mussen PH, Series editor. Flavell JH, Markman EM, Vol. editors. Handbook of child psychology: Cognitive development, 4th edn. Vol. 3. New York, NY: Wiley. pp 77–166

Chun, B. (2018, March). Personal interview.

Clarke, N. (2010). Emotional intelligence and its relationship to transformational leadership and key project manager competencies. Project Management Journal, 41(2), 5-20. doi:10.1002/pmj.20162

D'Mello, S. K. (2017). *Handbook of Learning Analytics* (First, pp. 115–127). Society for Learning Analytics Research (SoLAR). https://doi.org/10.18608/hla17.010

Darwin, C. R. 1872. The expression of the emotions in man and animals. London: John Murray. First edition. (n.d.). Retrieved April 18, 2018, from http://darwin-online.org.uk/content/frameset?pageseq=1&itemID=F1142&viewtype=text

Dhall, A., Goecke, R., Lucey, S., & Gedeon, T. (2011, November). Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In *Computer*

*Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on* (pp. 2106-2112). IEEE.

Dillon, J., Bosch, N., & Chetlur, M. (n.d.). Student Emotion, Co-occurrence, and Dropout in a MOOC Context, 5.

Duncan, D., Shine, G., & English, C. Facial emotion recognition in real time.

Dykstra, K., Whitehill, J., Salamanca, L., Lee, M., Carini, A., Reilly, J., & Bartlett, M. (2012). Modeling one-on-one tutoring sessions. In *Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on* (pp. 1–2). IEEE.

Eigen, D., & Fergus, R. (2015). Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2650-2658).

Ekman, P., & Rosenberg, E. (2005). *What the face reveals : Basic and applied studies of spontaneous expression using the facial action coding system (FACS)*(2nd ed. ed., Series in affective science). Oxford: Oxford University Press.

Fredrickson, B. L., & Branigan, C. (2005). Positive emotions broaden the scope of attention and thought-action repertoires. *Cognition & Emotion*, *19*(3), 313–332. https://doi.org/10.1080/02699930441000238

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning (adaptive computation and machine learning series). *Adaptive Computation and Machine Learning series*, 800.

Han, S., Pool, J., Tran, J., & Dally, W. (2015). Learning both weights and connections for efficient neural network. In *Advances in neural information processing systems* (pp. 1135-1143).

Hasanzadeh, R., & Shahmohamadi, F. (2011). Study of Emotional Intelligence and Learning Strategies. *Procedia - Social and Behavioral Sciences*, *29*, 1824–1829. https://doi.org/10.1016/j.sbspro.2011.11.430

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, *18*(7), 1527-1554.

Izard, C. E. (2010). The Many Meanings/Aspects of Emotion: Definitions, Functions, Activation, and Regulation. *Emotion Review*, *2*(4), 363–370. https://doi.org/10.1177/1754073910374661

Kaya, H., Gürpınar, F., & Salah, A. A. (2017). Video-based emotion recognition in the wild using deep transfer learning and score fusion. *Image and Vision Computing*, *65*, 66-75.

Keller, M. M., Chang, M.-L., Becker, E. S., Goetz, T., & Frenzel, A. C. (2014). Teachers' emotional experiences and exhaustion as predictors of emotional labor in the classroom: an experience sampling study. *Frontiers in Psychology*, *5*. https://doi.org/10.3389/fpsyg.2014.01442

Kosslyn, S. M. (in press). The science of learning. In S. M. Kosslyn & B. Nelson (Eds.), *Working universities: Minerva and the future of higher education.* Cambridge, MA: MIT Press. Retrieved from https://course-resources.minerva.kgi.edu/uploaded_files/mke/YRpz1r/chapter11-science-learning.pdf

Kosslyn, S. M. (in press). The science of learning. In S. M. Kosslyn & B. Nelson (Eds.), Working universities: Minerva and the future of higher education. Cambridge, MA: MIT Press. Retrieved from https://course-resources.minerva.kgi.edu/uploaded_files/mke/YRpz1r/chapter11-science-learning.pdf

Kosslyn, S. M., Kievit, R. A., Russell, A. G., & Shephard, J. M. (2012). PowerPoint® presentation flaws and failures: a psychological analysis. *Frontiers in Psychology* 3: 230.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

Larochelle, H., Bengio, Y., Louradour, J., & Lamblin, P. (2009). Exploring strategies for training deep neural networks. *Journal of machine learning research*, *10*(Jan), 1-40.

LeCun, Y., Bottou, L., Orr, G. B., & Müller, K. R. (1998). Efficient backprop. In *Neural networks: Tricks of the trade* (pp. 9-50). Springer, Berlin, Heidelberg.

Levi, G., & Hassner, T. (2015, November). Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. In *Proceedings of the 2015 ACM on international conference on multimodal interaction* (pp. 503-510). ACM.

Linnenbrink-Garcia, L., & Pekrun, R. (2011). Students' emotions and academic engagement: Introduction to the special issue. *Contemporary Educational Psychology*, *36*(1), 1–3. https://doi.org/10.1016/j.cedpsych.2010.11.004

Marton, F. (2006). Sameness and difference in transfer. *The Journal of the Learning Sciences*, *15*(4), 499-535.

neural networks - Which activation function for output layer? - Cross Validated. (n.d.). Retrieved April 20, 2018, from https://stats.stackexchange.com/questions/218542/which-activation-function-for-output-layer

McAllister, K. (2018, March). Personal interview.

Nielsen, M. A. (2015) How the backpropagation algorithm works. In *Neural Networks and Deep Learning (Chapter 2)*. Determination Press. Available at http://neuralnetworksanddeeplearning.com/chap2.html

Ohman, A., & Soares, J. J. (1994). "Unconscious anxiety": phobic responses to masked stimuli. *Journal of Abnormal Psychology*, *103*(2), 231–240.

Osterman, K. F. (2000). Students' Need for Belonging in the School Community. *Review of Educational Research*, *70*(3), 323–367. https://doi.org/10.3102/00346543070003323

Ouellet, S. (2014). Real-time emotion recognition for gaming using deep convolutional network features. ArXiv:1408.3750 [Cs]. Retrieved from http://arxiv.org/abs/1408.3750

Pardos, Z. A., Baker, R. S. J. D., San Pedro, M. O. C. Z., Gowda, S. M., & Gowda, S. M. (2013). Affective States and State Tests: Investigating How Affect Throughout the School Year Predicts End of Year Learning Outcomes. In *Proceedings of the Third International Conference on Learning Analytics and Knowledge* (pp. 117–124). New York, NY, USA: ACM. https://doi.org/10.1145/2460296.2460320

Pedro, M.O., Baker, R.S., Bowers, A.J., & Heffernan, N.T. (2013). Predicting College Enrollment from Student Interaction with an Intelligent Tutoring System in Middle School. *EDM*.

Pekrun, R., & Linnenbrink-Garcia, L. (2012). Academic emotions and student engagement. In *Handbook of Research on Student Engagement* (pp. 259-282). Springer US. DOI: 10.1007/978-1-4614-2018-7_12

Pekrun, R., Lichtenfeld, S., Marsh, H. W., Murayama, K., & Goetz, T. (2017). Achievement Emotions and Academic Performance: Longitudinal Models of Reciprocal Effects. *Child Development*, *88*(5), 1653–1670. https://doi.org/10.1111/cdev.12704

Pekrun's PDF: Pekrun, R. (n.d.). Achievement Emotions: Functions, Origins, and Implications for Practice, 145.

Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.

Ruthig, J. C., Perry, R. P., Hladkyj, S., Hall, N. C., Pekrun, R., & Chipperfield, J. G. (2008). Perceived control and emotions: interactive effects on performance in achievement settings. *Social Psychology of Education*, *11*(2), 161–180. https://doi.org/10.1007/s11218-007-9040-0

Salomon, G., & Globerson, T. (1987). Skill may not be enough: The role of mindfulness in learning and transfer. *International Journal of Educational Research*, *11*(6), 623-637.

Scholkopf, B., & Smola, A. J. (2001). *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press.

Schwarz, N. (2012). Feelings-as-Information Theory. In *Handbook of Theories of Social Psychology: Volume 1* (Vols. 1–1, pp. 289–308). London: SAGE Publications Ltd. https://doi.org/10.4135/9781446249215

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015, June). Going deeper with convolutions. Cvpr.

Thorndike, E. L. (1901). The human nature club: An introduction to the study of mental life. Longmans, Green and Company.

Wang, Z., Lukowski, S. L., Hart, S. A., Lyons, I. M., Thompson, L. A., Kovas, Y., … Petrill, S. A. (2015). Is Mathematical Anxiety Always Bad for Math Learning: The Role of Math Motivation. *Psychological Science*, *26*(12), 1863–1876. https://doi.org/10.1177/0956797615602471

Yi, D., Lei, Z., Liao, S., & Li, S. Z. (2014). Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*.

You, J. W., & Kang, M. (2014). The role of academic emotions in the relationship between perceived academic control and self-regulated learning in online learning. *Computers & Education*, *77*, 125–133. https://doi.org/10.1016/j.compedu.2014.04.018

Zimrot, R., & Ashkenazi, G. (2007). Interactive lecture demonstrations: a tool for exploring and enhancing conceptual change. *Chemistry Education Research and Practice*, *8*(2), 197-211.

Steyn, H. S. (2002). Practically significant relationships between two variables. SA Journal of Industrial Psychology, 28(3). doi:10.4102/sajip.v28i3.63

Cohen, J (1988). Statistical power analysis for the behavioural sciences. Second edition. Hillsdale, NJ: Lawrence Erlbaum Associates

Pekrun, R., Goetz, T., Frenzel, A. C., Barchfeld, P., & Perry, R. P. (2011). Measuring emotions in students' learning and performance: The Achievement Emotions Questionnaire (AEQ). Contemporary Educational Psychology, 36(1), 36-48. doi:10.1016/j.cedpsych.2010.10.002