Context-Aware Analysis of Group Submissions for Group Anomaly Detection and Performance Prediction

Narges Norouzi[1, 2], **Amir Mazaheri** [3]
[1] University of California, Berkeley
[2] University of California, Santa Cruz
[3] University of Central Florida
norouzi@berkeley.edu, amirmazaheri@knights.ucf.edu

Abstract

Learning exercises that activate students' additional cognitive understanding of course concepts facilitate contextualizing the content knowledge and developing higher-order thinking and problem-solving skills. Student-generated instructional materials such as course summaries and problem sets are amongst the instructional strategies that reflect active learning and constructivist philosophy.

The contributions of this work are twofold: 1) We introduce a practical implementation of an inside-outside learning strategy in an undergraduate deep learning course and will share our experiences in incorporating student-generated instructional materials learning strategy in course design, and 2) We develop a context-aware deep learning framework to draw insights from the student-generated materials for (i) Detecting anomalies in group activities and (ii) Predicting the median quiz performance of students in each group. This work opens up an avenue for effectively implementing a constructivist learning strategy in large-scale and online courses to build a sense of community between learners while providing an automated tool for instructors to identify at-risk groups.

Introduction

Artificial Intelligence (AI), Machine Learning (ML), and Data Science (DS) are emerging disciplines that combine mathematics, statistics, and Computer Science (CS) tools to develop methodologies that can extract knowledge from data to support discovery and decisions (Provost and Fawcett 2013). AI, ML, and DS have broad applications, including technology and sciences (natural and social), and have the potential to offer meaningful and welcoming pathways to Science, Technology, Engineering, and Mathematics (STEM) (Lue 2019; Donoho 2017; Lunch et al. 2019; Paleco et al. 2021).

In the past 50 years, the global demand for the adoption of AI and its democratization increased students' interest in enrolling in AI and ML-related courses at the graduate and undergraduate levels (National Academies of Sciences Engineering Medicine 2018; Duncan, Eicher, and Joyner 2020). For the successful implementation of AI education courses, specifically designed to serve a diverse set of students, a careful combination of computational and inferential thinking strategies is needed (Adhikari, DeNero, and Jordan 2021). Interdisciplinary and integrative fields like DS often require transformation and innovation in educational design practices (August et al. 2010). In doing so, innovative teaching strategies can effectively enable students' conceptual, procedural, and cognitive understanding of course concepts (Ramirez-Velarde et al. 2016). A study of National Science Foundation (NSF) proposals written by predominantly Science and Engineering faculty has shown that optimal learning outcomes for innovative and interdisciplinary programs include disciplinary grounding, integration, teamwork, communication, and critical awareness (Borrego and Newswander 2010). With these insights in mind, we develop and integrate new teaching paradigms into an undergraduate Deep Learning (DL) course to transform and improve student's learning experiences and outcomes. By doing so, we would meaningfully contribute to the body of evidence recommending effective pedagogies for inclusive teaching of AI, ML, and DS courses.

Student-Generated Instructional Materials

The Integrated Learning Process (ILP) model is based on the repetition of four stages of learning: 1) concrete experience, 2) construction, 3) abstraction, and 4) action (Ramirez-Velarde et al. 2014). Past studies have shown that ILPs that incorporate tasks requiring additional cognitive efforts such as retrieval practice, elaboration, concept mapping, or drawing encourage students to learn concepts on a deeper level and generate additional memory traces that aid with retention (Adesope, Trevisan, and Sundararajan 2017; Ebersbach, Feierabend, and Nazari 2020; Fernandes, Wammes, and Meade 2018).

One learning exercise that activates students' additional cognitive understanding of course concepts is student-generated instructional materials. As the term implies, student-generated instructional materials ask learners to provide objects that other students can use in their learning (Coppola and Pontrello 2020). When students generate instructional materials, it positively affects their learning through increased engagement (Collis and Moonen 2006), improved performance, and long-term retention of knowledge (King 1992). Types of student-generated instructional materials include lecture summaries (King 1992), problem-sets (Chin and Brown 2002), wiki (Ellis and Folley 2010),

video (Jordan et al. 2016), direct instruction of others (Alaimo et al. 2010; Hickey and Pontrello 2016), storyboards (Colbran and Gilding 2014), and flashcards (Colbran et al. 2017).

Student-generated problems are one type of student-generated instructional material. It is perceived as a paradigm shift from cognitive theory to constructivism, where students use their understanding of concepts to generate problems that bring about their cognitive development. The student-generated problems instructional method is developed as a part of the instructional strategies that reflect active learning and constructivist philosophy. This approach is an instructional strategy for helping students contextualize content knowledge and develop higher-order thinking and problem-solving skills (Algarni 2021).

Another type of student-generated instruction strategy is Inside-Outside (I-O) learning strategy which intends to emphasize transformations: from inside to outside and from learning to teaching (Yamamoto and Karaman 2011). In the I-O learning strategy, students collaborate to create their learning materials. Students develop memorization questionnaires, discussion topics and questions, schema and synoptic tables, relation trees, and other visualization aids. Sia (2015) shows that these cooperative and collaborative learning exercises will improve intercultural competencies (Schilstra, Takács, and Abcouwer 2019).

Contributions

The contributions of this work are twofold: 1) We introduce a practical implementation of an inside-outside learning strategy in an undergraduate deep learning course and will share our experiences in incorporating student-generated instructional materials learning strategy in course design, and 2) We develop a context-aware deep learning framework to draw insights from the student-generated materials for (i) Detecting anomalies in group activities and (ii) Predicting the median quiz performance of students in each group. This work opens up an avenue for effectively implementing a constructivism learning strategy in AI, ML, and DS courses while providing instructors with an automation tool to monitor group activities in large-scale courses.

Course Structure & Description of Data

This study is conducted in an undergraduate deep learning course at the University of California, Santa Cruz. The course provides a practical introduction to machine learning, with an emphasis on neural networks and deep learning. The course starts with a discussion of foundational pieces of statistical inference. Then, we introduce the basic elements of machine learning: loss functions and gradient descent. With these, we first present Logistic Regression (LR), or one-layer networks, and then move on to more complex models: deep neural networks, Convolutional Neural Networks (CNNs) for image recognition, Recurrent Neural Networks (RNNs), and Long-Short Term Memory (LSTM) for temporal and sequential data. The course also introduces unsupervised and self-supervised methods in the final two weeks. Collectively, the course teaches the core set of practical machine learning and deep learning techniques using real-world datasets.

The course is taught over a 10-week period with the schedule shown in Table 1.

Week	Topic	Due
1	Intro, Types of learning, & Feature Engineering	
2	Linear Regression, Least Squares, & Gradient Descent	Quiz 1 & Assignment 1
3	Regularization & Linear Classifiers	
4	Logistic Regression & Maximum Likelihood Estimation	Quiz 2 & Assignment 2
5	Introduction to Neural Networks	
6	Deep Learning topics: Dropout & Optimization	Quiz 3 & Assignment 3
7	Convolutional Neural Networks	
8	Recurrent Neural Networks	Quiz 4 & Assignment 4
9	Auto-Encoders & Intro to Self-Supervised Learning	
10	More on Self-Supervised Learning	Quiz 5 & Assignment 5

Table 1: Outline of the deep learning course over a 10-week period.

As shown above, the course does not assume any prior machine learning background and starts from basic data preparation and feature engineering topics, advances gradually into deep learning topics, and ends with self-supervision algorithms. Students work on bi-weekly assignments and take bi-weekly quizzes for course assessment.

The course was first designed and offered in the 2018-2019 academic year, and the demand for the course has grown over the years. Additionally, during the remote instruction of COVID-19, the instructional team looked into ways to engage students with the course content in a structured way. To this end, we use the following course activities:

- **Promoting collaborative learning**: The class is divided into 10 study groups (based on the idea of a mini-class introduced by Alvarado et al. in (Alvarado, Minnes, and Porter 2017)) so students can collaborate for course activities.
- **Incorporating student-generated instructional materials**: Teams of students are asked to submit weekly course summaries and problem sets. These materials are shared with all students in the course so they can use weekly problem sets and lecture summaries to prepare for bi-weekly quizzes.

Data is collected from the course offered in Fall 2021 from 100 students assigned to groups of 10. Out of 100 students, 18% are identified as women, 81% identified as men and 1% identified as non-binary.

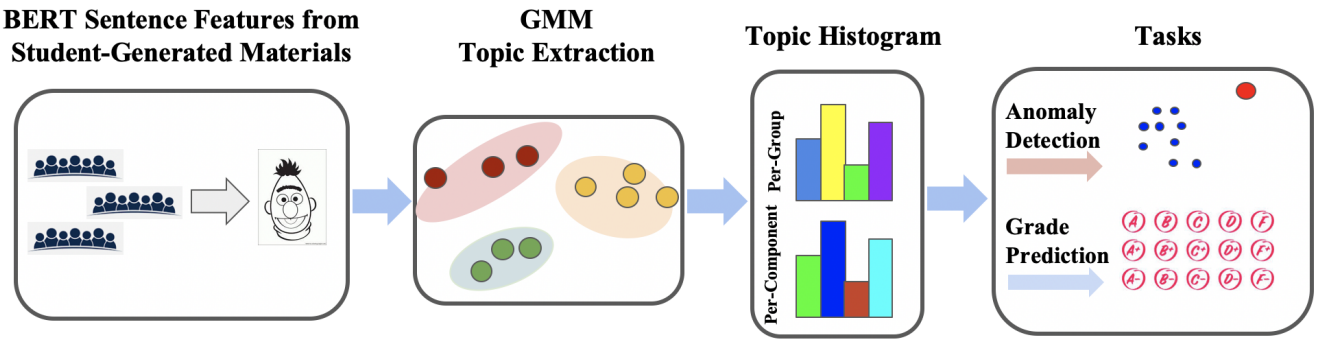

Figure 1: Student-generated materials, problem sets and lecture summaries, are passed to the S-BERT module to extract sentence embeddings. Gaussian Mixture Model (GMM) is used to classify sentence embeddings into different components. Per-component and per-group density features are then extracted from the GMM model and are used to 1) detect anomalies in group activities and 2) predict each group's median quiz grade.

Framework

We use the student-generated lecture summaries and problem sets to build a group anomaly detection module and the group's median quiz grade prediction module. Figure 1 presents the high-level design of our framework. The framework includes 1) A contextual embedding extraction of group summaries and problem sets using Sentence Bidirectional Encoder Representations from Transformers (S-BERT) (Reimers and Gurevych 2019; Devlin et al. 2018), 2) Gaussian Mixture Model (GMM) (Reynolds 2009) with eight components for topic extraction, 3) Anomaly detection module based on maximum Principle Component Analysis (PCA) reconstruction error, and 4) Prediction of group's median quiz grade. The following sections describe different elements of our framework.

Contextual Embedding

Finding a context-aware feature embedding from text is a fundamental research problem in Natural Language Processing (NLP). Different approaches are used in the literature for extracting features from the free-form educational text, such as Word2Vec (Mikolov et al. 2013), Glove (Pennington, Socher, and Manning 2014), and more recently, transformer-based model BERT. BERT and its variation S-BERT is used for many educational tasks such as analyzing surveys (Esmaeilzadeh et al. 2022), evaluation of students' essays and exams (Cochran et al. 2022; Padó 2022; Condor, Litster, and Pardos 2021), lecture summarization (Miller 2019), and evaluating lesson objectives (Cher, Lee, and Bello 2022).

In this work, group summaries and problem sets are passed to the pre-trained S-BERT to extract the 1024-dimensional sentence embeddings. To reduce noise in the data, PCA with 128 principle components is applied to problem sets and weekly summaries from all groups over ten weeks (100 documents for each submission type).

Gaussian Mixture Model for Topic Extraction

The Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions ($k \in \{1, 2, ..., K\}$) with unknown parameters. K is the number of components of our dataset. The GMM components are characterized by mean (μ), covariance matrix (Σ), and a mixing probability π that defines how big or small the Gaussian function will be. The mixing coefficients are themselves probabilities and must meet the following condition:

$$\sum_{k=1}^{K} \pi_k = 1. \tag{1}$$

GMM is trained on 128-dimensional sentence embeddings for problem sets and lecture summaries for group submissions over two weeks (corresponding to lectures associated with each quiz). For each component, we choose the point with maximal similarity to the mean (μ_k) of each Gaussian distribution as the representation of each component.

GMM models trained on weekly student submissions are used to extract features that can be used for downstream tasks. In this study, the number of Gaussian components used is experimentally chosen as 8 to capture as many weekly topics as possible while limiting the amount of variations and noise. We further investigate the choice of the number of components in the Discussion Section. Gaussian components are shown with C_k with $k \in \{1, ..., 8\}$. Assuming that the component C_k has N_{C_k} sentences and group G_i has a total of N_{G_i} sentences, we calculate the following density features:

Per-group Density For each group G_i with $i \in \{1, 2, ..., 10\}$, assume the number of sentences from group G_i that are assigned to components C_1 through C_8 are $\{N_1^{(G_i)}, N_2^{(G_i)}, ..., N_8^{(G_i)}\}$. We calculate per-group density feature vector for group G_i, $\rho_G(i)$, as:

$$\rho_G(i) = [\frac{N_1^{(G_i)}}{N_{G_i}}, \frac{N_2^{(G_i)}}{N_{G_i}}, ..., \frac{N_8^{(G_i)}}{N_{G_i}}] \tag{2}$$

$\rho_G(i) \in \mathbb{R}^8$ is a feature vector measuring the concentration or density of the submission from a group G_i in each of

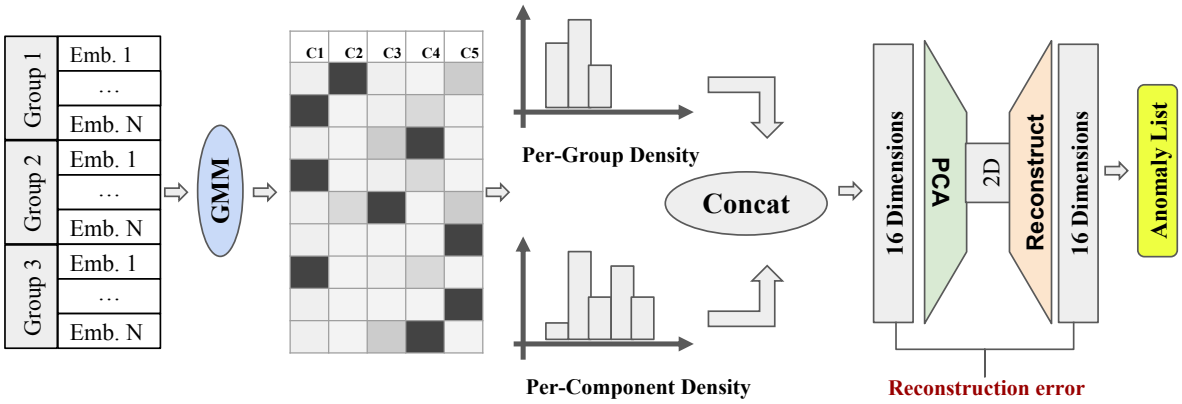

Figure 2: The outline of the anomaly detection module is illustrated. The module is made up of 1) S-BERT feature extraction, 2) Topic extraction using GMM, 3) Feature extraction (per-group and per-component densities), 4) $16D \rightarrow 2D$ PCA dimensionality reduction and $2D \rightarrow 16D$ reconstruction, and 5) Anomaly detection based on the reconstruction error.

the Gaussian components. Imbalance concentrations in each component C_k compared to other groups' densities can be an indication of an anomaly in the submission.

Per-component Density Assuming that the number of sentences assigned to components C_1 through C_8 are $N_{C_1}, N_{C_2}, ..., N_{C_8}$, we define per-component density for group G_i, $\rho_C(i)$ as follows:

$$\rho_C(i) = [\frac{N_1^{(G_i)}}{N_{C_1}}, \frac{N_2^{(G_i)}}{N_{C_2}}, ..., \frac{N_8^{(G_i)}}{N_{C_8}}] \tag{3}$$

$\rho_C(i) \in \mathbb{R}^8$ is a feature vector measuring the percentage of sentences in each of the 8 components that are associated with group G_i. Large deviations from 10% contribution (based on uniform distribution of sentences between groups) to each component can indicate group submission anomaly.

Group Anomaly Detection

For each group submission, feature vectors are created using the concatenation of per-group density vector, $\rho_G(i) \in \mathbb{R}^8$, and per-component density vector, $\rho_C(i) \in \mathbb{R}^8$. Therefore, feature vectors for group G_i's problem set and lecture summary for week j, $f_G^{(p_j)}(i)$ and $f_G^{(s_j)}(i)$, are defined as follows.

$$f_G^{(p_j)}(i) = [\rho_G^{(p_j)}(i), \rho_C^{(p_j)}(i)] = [\frac{N_1^{(G_i,p_j)}}{N_{G_i,p_j}}, \frac{N_2^{(G_i,p_j)}}{N_{G_i,p_j}}, ..., \frac{N_8^{(G_i,p_j)}}{N_{G_i,p_j}}, \frac{N_1^{(G_i,p_j)}}{N_{C_1,p_j}}, \frac{N_2^{(G_i,p_j)}}{N_{C_2,p_j}}, ..., \frac{N_8^{(G_i,p_j)}}{N_{C_8,p_j}}] \tag{4}$$

$$f_G^{(s_j)}(i) = [\rho_G^{(s_j)}(i), \rho_C^{(s_j)}(i)] = [\frac{N_1^{(G_i,s_j)}}{N_{G_i,s_j}}, \frac{N_2^{(G_i,s_j)}}{N_{G_i,s_j}}, ..., \frac{N_8^{(G_i,s_j)}}{N_{G_i,s_j}}, \frac{N_1^{(G_i,s_j)}}{N_{C_1,s_j}}, \frac{N_2^{(G_i,s_j)}}{N_{C_2,s_j}}, ..., \frac{N_8^{(G_i,s_j)}}{N_{C_8,s_j}}] \tag{5}$$

In the equations above,

- $N_d^{(G_i,s_j)}$: The number of sentences in the group G_i's lecture summary submission that are associated with component C_d.
- N_{G_i,s_j}: The total number of sentences in group G_i's lecture summary submission.
- N_{C_d,s_j}: The total number of sentences associated with component C_d from all group's lecture summary submissions in week j.
- $N_d^{(G_i,p_j)}$, N_{G_i,p_j}, and N_{C_d,p_j} follow the same definition as the first three items but for groups' problem set submissions.

We identify anomalous submissions using PCA reconstruction error. Specifically, we apply PCA to convert feature vectors $f_G^{(p_j)}(i)$ and $f_G^{(s_j)}(i)$ to 2-dimensional space. We calculate reconstruction error for all embeddings. Assuming that principle components are orthonormal vectors $\{v_1, v_2, ...v_k\}$, we calculate the reconstruction error as:

$$e_f = ||f - \sum_{i=1}^{k} w_i v_i||_2 \quad \text{where} \quad w_i = v_i \cdot f \tag{6}$$

Submissions with reconstruction errors higher than 2 standard deviations from the mean of the reconstruction errors are labeled as a potential anomaly. Figure 2 illustrates an overview of the anomaly detection module.

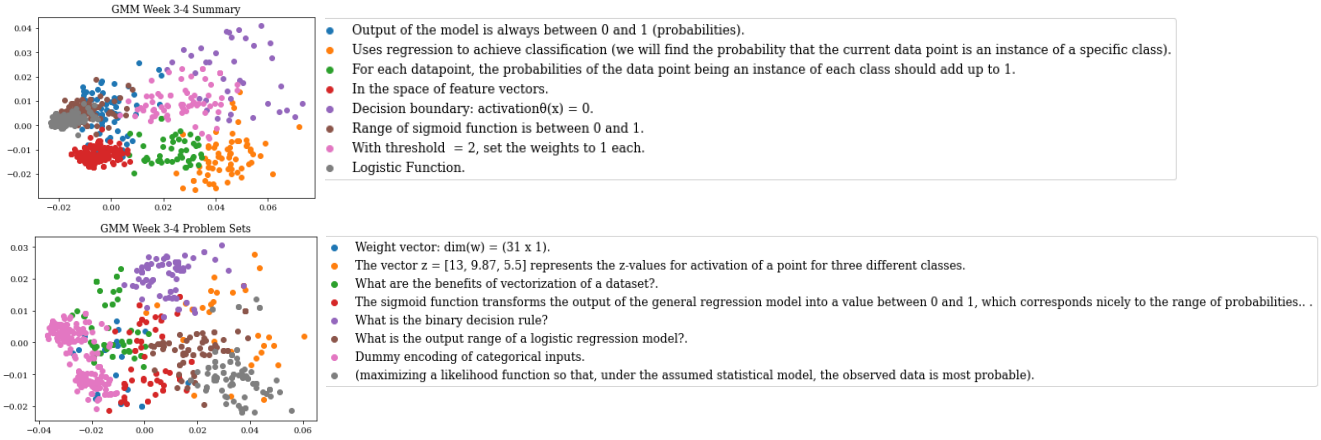

Figure 3: The top figure shows GMM components for lecture summaries for weeks 3 and 4. The bottom figure shows GMM components for problem sets for weeks 3 and 4. As shown, the component frequencies and topics remain similar and cover topics discussed in the course during weeks 3 and 4 (logistic regression and regularization).

Topic	**Component centers**
Regularization	**(C2)** "Generally, we don't want huge weights." **(C5)** "If the dataset is small, the probability of having a complex model overfitting to the data model is going to increase."
Linear classification	**(C3)** "With threshold=2 set the weights to 1 each." **(C4)** "In the space of feature vectors."
Logistic regression	**(C1)** "Maps points in d-dimensional space into to value in range (0,1)." **(C6)** "Logistic Function" **(C8)** "Using the logistic function, can map activation score z to range (0, 1)."
Max likelihood estimation	**(C7)** "Decision boundary is $activation(\theta x) = 0$."

Table 2: Association of 8 GMM components (C_1 through C_8) to topics covered in weeks 3 and 4 based on lecture summary submissions.

Group Grade Prediction

Using the top-2 principle components obtained in the previous step, we train a linear regression model to predict each group's median bi-weekly quiz score based on lecture summaries and problem sets submitted over the prior two weeks. The benefit of the median quiz prediction is to augment the anomaly detection module and distinguish between underperforming and over-performing groups. This enables the framework to triage some anomaly cases as over-performing submissions.

Results

In this section, we outline the framework's performance in topic extraction from weekly lecture summaries and problem sets using GMM on BERT embeddings of sentences. We will also discuss the framework's success in identifying anomalous group submissions for identifying at-risk groups.

Quality of Topic Extraction Module

Figure 3 illustrates 2-dimensional PCA plots of the two GMM models with eight components trained on the submission in weeks 3 and 4 for lecture summaries (top figure) and problem sets (bottom figure). In both cases, sentences with the minimum distance to the center of the 8 Gaussian distributions are used as representative sentences for each component. As shown, the probability density distribution of lecture summaries and problem sets are similar, suggesting that the density of groups' attention to the core topics in the lecture summaries and problem sets are aligned.

It is also important to note that during weeks 3 and 4, as shown in Table 1, regularization, linear classification, logistic regression, and maximum likelihood estimation topics were covered. Tables 2 and 3 outline the association of Gaussian components (represented by their centers) to each of these four core topics in weeks 3 and 4 based on the lecture summaries and problem set submissions, respectively.

Performance of Anomaly Detection Module

Table 4 outlines groups' median quiz grades for quizzes 1 through 5 and results from the anomaly detection module for weeks 1-2 through 9-10. It is important to note that not all submissions flagged as anomalies belong to underper-

Topic	Component centers
Regularization	**(C3)** "What are the benefits of vectorization of a dataset?"
Linear classification	**(C1)** "weight vector: $dim(w) = 31 \times 1$." **(C2)** "The vector $z = [13, 9.87, 5.5]$ represents the z-value for activation of a point for three different classes." **(C7)** "Dummy encoding of categorical inputs."
Logistic regression	**(C4)** "The sigmoid function transforms the output of the general regression model into a value between 0 and 1, which corresponds nicely to the range of probabilities." **(C5)** "What is the binary decision rule?" **(C6)** "What is output range of the logistic regression model?"
Max likelihood estimation	**(C8)** "Maximizing a likelihood function so that, under the assumed statistical model, the observed data is more probable"

Table 3: Association of 8 GMM components (C_1 through C_8) to topics covered in weeks 3 and 4 based on problem set submissions.

Group	Bi-Weekly Median Quiz Grade					Flagged as Outlier				
	Q1	Q2	Q3	Q4	Q5	Weeks 1-2	Weeks 3-4	Weeks 5-6	Weeks 7-8	Weeks 9-10
1	17.5	18.75	13.92	13.705	13.75	Yes			Yes	
2	19.0	18.0	20.33	20.5	19.0			Yes		Yes
3	22.0	19.0	22.5	18.33	19.75					
4	21.0	20.0	22.0	19.83	21.0					
5	17.0	19.0	19.58	15.665	18.75					
6	15.125	14.5	16.625	18.17	14.25					
7	17.0	17.75	18.455	14.295	20.25		Yes		Yes	
8	17.5	16.375	19.33	14.5	15.0	Yes				Yes
9	19.0	20.5	21.33	18.33	16.0					
10	16.0	17.0	19.33	14.83	17.0					

Table 4: A summary of groups' median quiz grades for quizzes 1 through 5 and results from the anomaly detection module for corresponding weeks.

forming groups. In some cases, though rarely, we have observed that more detailed submissions might also get flagged as anomalies. The framework is augmented with the next module, grade prediction, to help distinguish between over-performing and under-performing groups and triage these cases.

The anomaly detection module reduces the time instructors need to check lecture summaries and problem set submissions. Identifying underperforming groups in a timely manner provides an opportunity for the instructional team to identify at-risk teams and intervene if needed.

In the analysis of submissions from the groups flagged by the anomaly detection module, 16% of submissions are flagged as anomalies, of which 63.5% belong to underperforming submissions (10% of the total submissions). To further examine anomaly cases for weeks 1 and 2, we identify the following:

- Group 8 is identified as over-performing as they included a very detailed summary and questions in their submission. This group can be triaged using the median grade prediction module.
- Group 1 failed to submit a quality summary as the submission had only a bulleted list of topics discussed in the course. Group 1's problem set submission only included a maximum of two questions per lecture, which the model might have picked as an anomaly as it does not contribute equally to the density of the GMM components.

As another example, in the submissions for weeks 5 and 6, group 2 is flagged as an anomaly. Looking into the group's median quiz grade, the group has a higher median than the average of the other groups (the same is also predicted using the grade prediction module). Therefore we can conclude that the group might have been over-performing. This has been verified by checking the group's submission for the corresponding weeks.

Discussion

We train GMM models from $K = 4$ components to $K = 10$. We chose 4 as the lower bound to ensure that the main bi-weekly covered topics are assigned to different components. We chose $K = 10$ as the upper bound to avoid creating a high-dimensional output from the GMM model and hence overfitting the training data for the downstream tasks.
We calculated Akaike Information Criterion (AIC) scores to identify the appropriate number of components. As shown

Sentences
"Project an input x on theta resulting in activation z (real valued)."
"Map z to $(0, 1)$ by using the logistic function ."
"Weighted sum of linear classifier output is activation (determines classification).""
"Output of model is $0 - 1$ (Probability)."
"$p(C = 1\|x) = sigmoid() =$".
"Vector in d+1 dimensional feature space."
"Consider model: $h_theta(x) = \theta_j \times x_j$."
"$X \in \mathbb{R} \times (d + 1)$."
"Maps point in d-dimensional space to value in range $(0, 1)$."
"Maps point in (d+1)-dimensional space to real $\#$."
"Output of the model is always between 0 and 1 (probabilities)."
"$x \rightarrow \phi(x) \rightarrow y(outcome)$."
"Map z to the range 0 to 1 using the logistic function (sigmoid function)."
"The output of Logistic Regression is always a value between (0,1) to form a probability distribution."
"range(-inf, +inf) while domain (0,1)."

Table 5: A sample of sentences associated with component C_0 for lecture summaries in weeks 3 and 4 represents the logistic function's range and domain.

in Figure 4, AIC scores drop sharply at $K = 8$, and hence we conclude that 8 components are appropriate.

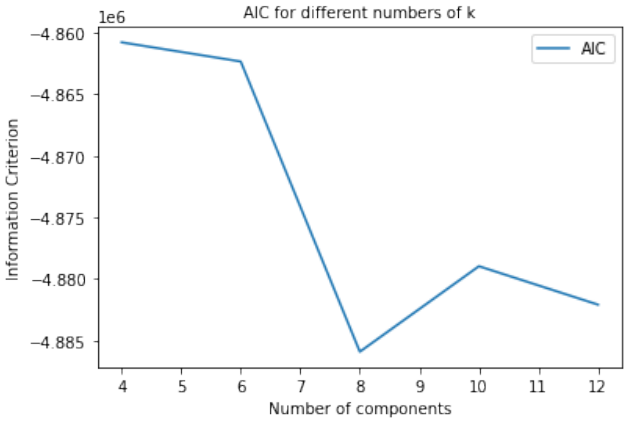

Figure 4: AIC score for different number of components of the GMM model trained on the entire training data.

Qualitative evaluation of the semantic coherence of components Furthermore, we qualitatively evaluated the semantic coherency of components in each of the GMM models. Table 5 lists a sample of 15 sentences from component C_1 of the GMM trained on lecture summaries from weeks 3 and 4. As shown, the sentences in component C_1 are coherently related to each other and describe the range and domain of the logistic function.

Performance of Grade Prediction Module

We trained the model on the density feature vectors after applying a 2D PCA. We randomly selected 90% of feature vectors as training and 10% as the test set. The average mean squared error between the group's median quiz grade and the model's prediction is 1.16 out of 25 (maximum quiz grade).

It is important to note that the sample size in the current dataset is limited. However, as the course is offered for more quarters and historical data is gathered, the model can be retrained using all historical data to enhance feature learning and improve the quality of the regression model.

Limitations & Future Work

The sample size for the current study is 100 weekly lecture summary submissions and 100 weekly problem sets. Though the anomaly detection and grade prediction modules show promise in detecting over-performing and under-performing group submissions, more data can improve the performance of both modules.

Future work can include a DL framework for classifying anomalies as over-performing or under-performing to more accurately separate the two groups. Additionally, using more textual data and problem sets from DL topics can enable us to fine-tune the S-BERT module to obtain more context-relevant sentence embeddings.

Conclusion

Summarization capability and the ability to create problem sets from the materials learned in a course are among the factors that indicate students' mastery of a topic. In this paper, we introduce and validate a pedagogically effective group activity in a deep learning course to improve students' mastery of deep learning concepts through writing weekly course summaries and problem sets. We employ a context-aware deep learning framework for analyzing group summaries and problem sets to monitor group dynamics. The framework can also be used to predict groups' quiz performance while also providing an automated tool to instructors an avenue for identifying at-risk groups.

References

Adesope, O. O.; Trevisan, D. A.; and Sundararajan, N. 2017. Rethinking the use of tests: A meta-analysis of practice testing. *Review of Educational Research*, 87(3): 659–701.

Adhikari, A.; DeNero, J.; and Jordan, M. I. 2021. Interleaving Computational and Inferential Thinking: Data Science for Undergraduates at Berkeley. *arXiv preprint arXiv:2102.09391*.

Alaimo, P. J.; Langenhan, J. M.; Tanner, M. J.; and Ferrenberg, S. M. 2010. Safety teams: An approach to engage students in laboratory safety. *Journal of Chemical Education*, 87(8): 856–861.

Algarni, S. 2021. *Using Student-Generated Problems (Sgp) as an Instructional Strategy to Enhance Undergraduate Engineering Students' Knowledge Application Ability and Problem-Solving Skills*. Ph.D. thesis, The University of North Dakota.

Alvarado, C.; Minnes, M.; and Porter, L. 2017. Microclasses: A structure for improving student experience in large classes. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, 21–26.

August, P.; Swift, J.; Kellogg, D.; Page, G.; Nelson, P.; Opaluch, J.; Cobb, J.; Foster, C.; and Gold, A. 2010. The T assessment tool: a simple metric for assessing multidisciplinary graduate education. *Journal of Natural Resources and Life Sciences Education*, 39(1): 15–21.

Borrego, M.; and Newswander, L. K. 2010. Definitions of interdisciplinary research: Toward graduate-level interdisciplinary learning outcomes. *The Review of Higher Education*, 34(1): 61–84.

Cher, P. H.; Lee, J. W. Y.; and Bello, F. 2022. Machine Learning Techniques to Evaluate Lesson Objectives. In *International Conference on Artificial Intelligence in Education*, 193–205. Springer.

Chin, C.; and Brown, D. E. 2002. Student-generated questions: A meaningful aspect of learning in science. *International Journal of Science Education*, 24(5): 521–549.

Cochran, K.; Cohn, C.; Hutchins, N.; Biswas, G.; and Hastings, P. 2022. Improving automated evaluation of formative assessments with text data augmentation. In *International Conference on Artificial Intelligence in Education*, 390–401. Springer.

Colbran, S.; and Gilding, A. 2014. Exploring legal ethics using student generated storyboards. *The Law Teacher*, 48(3): 296–320.

Colbran, S.; Gilding, A.; Colbran, S.; Oyson, M. J.; and Saeed, N. 2017. The impact of student-generated digital flashcards on student learning of constitutional law. *The Law Teacher*, 51(1): 69–97.

Collis, B.; and Moonen, J. 2006. The contributing student: Learners as co-developers of learning resources for reuse in web environments. In *Engaged learning with emerging technologies*, 49–67. Springer.

Condor, A.; Litster, M.; and Pardos, Z. 2021. Automatic Short Answer Grading with SBERT on Out-of-Sample Questions. *International Educational Data Mining Society*.

Coppola, B. P.; and Pontrello, J. K. 2020. Student-generated instructional materials. In *Active learning in college science*, 385–407. Springer.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Donoho, D. 2017. 50 years of data science. *Journal of Computational and Graphical Statistics*, 26(4): 745–766.

Duncan, A.; Eicher, B.; and Joyner, D. A. 2020. Enrollment motivations in an online graduate CS program: Trends & gender-and age-based differences. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, 1241–1247.

Ebersbach, M.; Feierabend, M.; and Nazari, K. B. B. 2020. Comparing the effects of generating questions, testing, and restudying on students' long-term recall in university learning. *Applied Cognitive Psychology*, 34(3): 724–736.

Ellis, C.; and Folley, S. 2010. Students writing their own lectures with a wiki and the CSA. In *Handbook of research on hybrid learning models: Advanced tools, technologies, and applications*, 244–259. IGI Global.

Esmaeilzadeh, S.; Williams, B.; Shamsi, D.; and Vikingstad, O. 2022. Providing Insights for Open-Response Surveys via End-to-End Context-Aware Clustering. *arXiv preprint arXiv:2203.01294*.

Fernandes, M. A.; Wammes, J. D.; and Meade, M. E. 2018. The surprisingly powerful influence of drawing on memory. *Current Directions in Psychological Science*, 27(5): 302–308.

Hickey, T. J.; and Pontrello, J. K. 2016. Building bridges between science courses using honors organic chemistry projects. *Journal of College Science Teaching*, 46(1): 18.

Jordan, J. T.; Box, M. C.; Eguren, K. E.; Parker, T. A.; Saraldi-Gallardo, V. M.; Wolfe, M. I.; and Gallardo-Williams, M. T. 2016. Effectiveness of student-generated video as a teaching tool for an instrumental technique in the organic chemistry laboratory. *Journal of Chemical Education*, 93(1): 141–145.

King, A. 1992. Comparison of self-questioning, summarizing, and notetaking-review as strategies for learning from lectures. *American Educational Research Journal*, 29(2): 303–323.

Lue, R. A. 2019. Data science as a foundation for inclusive learning. *Harvard Data Science Review*.

Lunch, C.; Crall, A.; Jones, M. A.; and Jones, K. D. 2019. Diversity, equity, and inclusion in data science: introducing the environmental data science inclusion network (EDSIN). In *AGU Fall Meeting Abstracts*, volume 2019, ED21E–1068.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Miller, D. 2019. Leveraging BERT for extractive text summarization on lectures. *arXiv preprint arXiv:1906.04165*.

National Academies of Sciences Engineering Medicine, N. 2018. *Data science for undergraduates: Opportunities and options*. National Academies Press.

Padó, U. 2022. Assessing the Practical Benefit of Automated Short-Answer Graders. In *International Conference on Artificial Intelligence in Education*, 555–559. Springer.

Paleco, C.; García Peter, S.; Salas Seoane, N.; Kaufmann, J.; Argyri, P.; et al. 2021. Inclusiveness and diversity in citizen science. *The science of citizen science*, 261.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.

Provost, F.; and Fawcett, T. 2013. Data science and its relationship to big data and data-driven decision making. *Big data*, 1(1): 51–59.

Ramirez-Velarde, R.; Alexandrov, N.; Sanhueza-Olave, M.; and Perez-Cazares, R. 2016. The impact of learning activities on the final grade in engineering education. *Procedia Computer Science*, 80: 1812–1821.

Ramirez-Velarde, R.; Perez-Cazares, R.; Alexandrov, N.; and Garcia-Rueda, J. J. 2014. Education 2.0: Student generated learning materials through collaborative work. *Procedia Computer Science*, 29: 1835–1845.

Reimers, N.; and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Reynolds, D. A. 2009. Gaussian Mixture Models. In *Encyclopedia of Biometrics*.

Schilstra, T.; Takács, E.; and Abcouwer, T. 2019. COOPERATIVE LEARNING IN A HIGHER EDUCATIONAL SETTINGRealizing high-performing cooperative learning in higher education. *Proceedings of the 2019 AIS SIGED International Conference on Information Systems Education and Research*.

Yamamoto, G. T.; and Karaman, F. 2011. Education 2.0. *On the Horizon*.