March 2026
Volume 17 Issue 1

JOCSE

Journal Of Computational Science Education

Promoting the Use of Computational Science Through Education

ISSN 2153-4136 (online)

Contents

Volume 17, Issue 1

Introduction to Volume 17, Issue 1

David Joiner, Editor
Kean University
djoiner@kean.edu

Issue 17-1 of the Journal of Computational Science Education features 12 articles, including 11 articles presented at the 12th SC Workshop on Best Practices for High Performance Computing and Education Workshop at SC25 in St Louis, MO.

The 12th annual BPHTE meeting featured many different themes, including AI as both subject and tool. Mullen et al. presented materials focused on training researchers to build and use LLM-RAG models, and Dey and Lindsey showcased work using narrative datasets to teach AI-ML pipelines. Morsy et al. employed AI agents to validate metadata in HPC Education digital library learning objects, and Rasul and Stuart studied user attitudes towards an AI Assistant designed to help guide users to answers in their HPC systems. Broadening participation in HPC is another theme represented here. Johnston et al. present an HPC education project aimed at cluster competitions engaging undergraduates from historically disadvantaged communities in South Africa. Yu et al. present introductory HPC training for users without prior technical background. Luchini-Colbry et al. showcase the CyberAmbassadors program, including mentor networks and a focus on culturally aware mentoring. Another theme was the use of competitions and experiential learning. Carbunaru and Sami describe cluster competition programs at the National University of Singapore, and Jezghani and Fry describe a physical sciences hackathon. Cross-institutional collaboration was also on display, with Diehl et al. presenting community driven curriculum development across the national labs, and Sukhija et al. presenting a national roadmap derived from the NAIRR Pilot User Experience Working Group. Additionally, JOCSE 17-1 features a student paper by Iyinbor et al. describing an application of machine learning to classify steel alloys by stacking fault energy regimes.

We encourage you to submit your work to the Journal of Computational Science Education. Computational science is an increasingly important interdisciplinary field, offering insights into complex systems, accelerating discovery, and helping to solve diverse problems. We welcome high-quality papers describing instructional materials, successful projects, or research on instructional efficacy. Whether you are faculty or a student, your contributions are valuable to advancing computational science education. Additionally, if you have expertise in computational science, consider volunteering as a reviewer to support our peer review process. Together, we can share successes and inspire others to develop and adopt computational science in education.

Sincerely, Dave Joiner

Machine Learning Prediction of Stacking Fault Energy in Steel Alloys Based on Chemical Composition

Ikponmwosa J. Iyinbor
University of Southern California
iyinbor@usc.edu

Ken-ichi Nomura
University of Southern California
knomura@usc.edu

Paulo S. Branicio
University of Southern California
branicio@usc.edu

ABSTRACT

Stacking fault energy (SFE) is a critical parameter in the design of steels with desirable mechanical properties such as strength, ductility, and strain-hardening rate. SFE influences secondary deformation mechanisms like Transformation Induced Plasticity (TRIP) and Twinning Induced Plasticity (TWIP). This work involves creating a machine learning model to classify steel alloys into low, medium, or high SFE categories, aiding in the prediction of secondary deformation behaviors. Data from literature containing experimental and theoretical SFE values for various steel alloy compositions were compiled and preprocessed, resulting in a dataset of 374 observations. Using this dataset, several machine learning models, including Feedforward Neural Network (FFNN), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest (RF), Gradient Boosting Regressor (GBR), and CatBoost Regressor (CAT), and Adaptive Boost Regressor (ADB) were trained and evaluated for SFE prediction accuracy. Two models, SVM and RF, emerged as the top-performing models. To enhance accuracy and reduce misclassification, threshold probabilities were applied, allowing fuzzy classification when model uncertainty was high. Validation against literature data showed strong agreement between predictions and reported SFE values. This study provides valuable insights into predicting SFE and guiding the development of austenitic steel alloys with tailored properties.

KEYWORDS

Supervised Learning, Stacking Fault Energy, Random Forest, Gradient Boosting, Adaptive Boost, Support Vector Machine, Neural Network

1 INTRODUCTION

A stacking fault is an interruption in the normal stacking sequence of atomic planes, and the energy required to create this interruption is known as the stacking fault energy (SFE). The SFE has great importance in designing steels with superior combination of mechanical strength, ductility and strain hardening rate. SFE plays an important role in activating secondary deformation mechanisms such as Transformation-induced and twinning-induced plasticity (TRIP, TWIP) observed in austenitic steels. Experimental work has shown that these deformation mechanisms are primarily a function of their SFE, such that with increasing SFE from low to high, the deformation mechanisms change from TRIP (< 20 mJ/m^2) to TWIP (between 20 mJ/m^2 – 45 mJ/m^2) and then to MBIP (> 45 mJ/m^2), which stands for Microband-Induced Plasticity as shown in figure 1 [2–5, 10, 11].

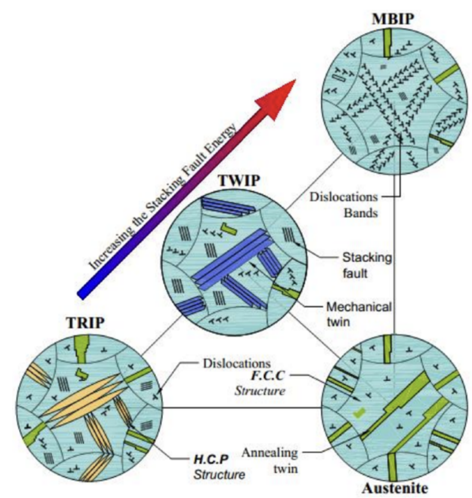


Figure 1: Schematic representation of secondary deformation mechanisms (TRIP, TWIP, and MBIP) in austenitic steels
Note: Reprinted with permission from [2].

Due to this relationship between SFE and secondary deformation mechanisms, knowing the SFE of a steel alloy is crucial in designing advanced structural materials [3, 5]. Several techniques, such as measuring the geometry of dislocations and the diffraction profile, have been used to compute SFE values [7, 9]. However, measuring exact SFE values can be challenging because there are different methods to calculate it, and each has its own sources of error, such as material constants used in the calculation or other physical and material interactions (strain in material) to name a few [3]. Regardless of these uncertainties in calculating exact SFE values, it is possible to classify austenitic steel alloys into different SFE regimes (low SFE (< 20 mJ/m^2), medium SFE (between 20 mJ/m^2 – 45 mJ/m^2), and High SFE (> 45 mJ/m^2). While exact SFE values can vary due to experimental errors or other factors, knowing the general SFE regime is enough to determine the secondary deformation mechanisms in austenitic steels [3, 4].

This project involves creating a machine learning model to predict the SFE regime for different steel alloy compositions. The training data for this model comes from a database of information from various research studies in literature that cover a range of steel

https://doi.org/10.22369/issn.2153-4136/17/1/1

alloy compositions, including both experimental and theoretical SFE calculations [3].

2 DATA PREPROCESSING AND VISUALIZATION

2.1 Data Collection and Classification

The dataset used in this study was obtained from a database curated by extensively gathering information from literature on various steel alloy compositions [9]. The data included results from experimental and theoretical Stacking Fault Energy (SFE) calculations based on these compositions. The focus was on elements commonly used in the design of austenitic steels, ensuring that our dataset represented a comprehensive range of these alloying elements. The complete list of elements considered for data collection included: C, N, P, S, V, Ni, Nb, Al, Ti, Fe, Hf, Mo, Mn, Co, Si, Cr, and Cu.

The SFE data came from various temperature ranges and was collected through different experimental techniques like Transmission Electron Microscopy (TEM), X-ray Diffraction (XRD), Neutron Diffraction, and thermodynamic modeling. This resulted in a dataset with about 500 data points, each representing a unique combination of material composition, temperature, and method used to obtain the SFE.

To prepare the data for analysis, we applied a systematic approach to data preprocessing. Initially, a subset of 426 observations was chosen, including only data collected at room temperature (300 K), because most of the data in the complete dataset were taken at this temperature, so it made sense to exclude temperature as a variable when training our models. Next, the subset was further refined by removing theoretical modeling data and keeping only the experimental measurements since these represent truth values for SFE. This reduced the subset to 387 observations. Data related to nickel-based alloys was removed, leaving only measurements from ferrous-based alloys, resulting in 379 observations. Lastly, another step involved filtering to ensure there were enough data points from each alloying element used in austenitic steels. Elements such as P, S, V, Ti, Hf, Nb, Co, and Cu had too few data points, so they were excluded from the training dataset. The final list of elements used as predictor variables in the training input data included C, N, Ni, Al, Fe, Mo, Mn, Si, and Cr. After all filtering steps, the final training dataset contained 374 observations. This final subset was then used for further analysis, providing a balanced and reliable data set for the study.

As mentioned in the introduction, understanding the general SFE regime is sufficient to predict deformation mechanisms. Therefore, as seen in Figure 2, each steel alloy was divided into three SFE categories: Low, Medium, and High, corresponding to the secondary deformation mechanisms TRIP, TWIP, and dislocation glide, respectively. The final training dataset contains 77 observations in the Low SFE class, 209 in the Medium SFE class, and 88 in the High SFE class.

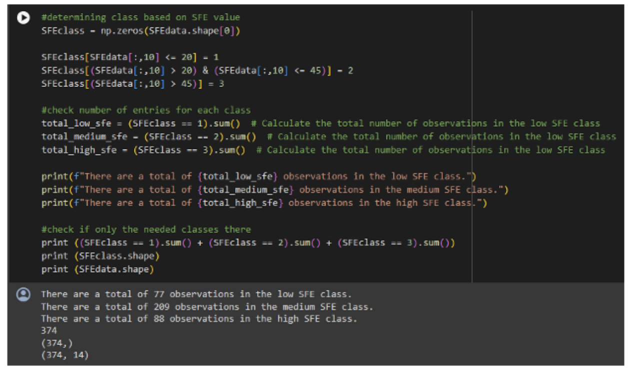


Figure 2: Classification of steel alloys into Low, Medium, or High SFE categories based on their respective SFE values.

2.2 Data Visualization and Dimensionality Reduction

Initially, the relationship between two elements, Ni and Cr, and their impact on SFE is visualized, as shown in Figure 3. The 3D plot reveals a non-linear correlation between SFE and these elements, indicating that the effect of increasing or decreasing either element is not straightforward. This suggests that the influence of their weight percentages (wt%) on SFE cannot be summarized into a simple rule, especially given the limited dataset used for training.

Next, the analysis explores how variations in wt% Cr and another alloying element impact SFE. Figure 4 also shows that no clear trend emerges from the interaction between Cr and other alloying elements, further confirming the complex, non-linear relationship between alloy composition and SFE. The data visualization in Figure 4 provides a limited view, as it examines only a few changing parameters at a time. To gain deeper insights into the relationship between SFE and alloy compositions, it's necessary to visualize the impact of varying all parameters simultaneously. However, with nine predictor variables (alloying elements) and SFE values resulting in a 10-dimensional dataset, spotting meaningful patterns becomes challenging. Dimensionality reduction techniques, such as Principal Component Analysis (PCA), are required to address this high dimensionality while retaining as much crucial information as possible.

PCA is used to compress the 10-dimensional dataset into a 2-D or 3-D solely for visual representation. To achieve this, two separate PCA transformations are applied to the normalized dataset. The first transformation reduces the data to 2 principal components, while the second transformation reduces it to 3 principal components. The 3-D PCA retains about 84.53% of the total variance in the data. The amount of variance explained by each principal component indicates that PC1 (52.33%) is the direction in the data that captures more than half the total variance in the dataset, while PC2 (24.18%) and PC3 (8.02%) explains the rest of the total variance. Additionally, the PCA loading scores for the three principal components are reported in Table 1, where features with higher absolute values contributes to loading scores defining the PC direction. For example, for the PC1 direction, the Mo, Cr, and Ni features are the three strongest positive contributors for this direction, while the Mn, Fe and C features have the three strongest negative influence.

Since the PCA performed in this project is solely for data visualization to identify patterns, this level of variance is sufficient for looking at the high-dimensional data in 2D and 3D.

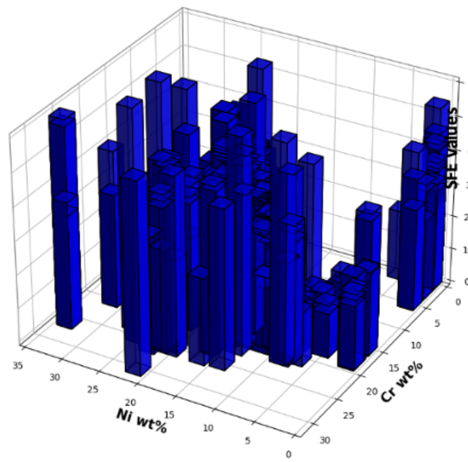


Figure 3: 3D bar plot showing the relationship between wt% Ni and wt% Cr on SFE.

Figure 5 presents the 3-D PCA transformation. This visualization doesn't reveal a clear pattern, as the SFE classes seem mixed without distinct clusters. To improve clarity, a 2-D visualization is generated, shown in Figure 6. This 2-D PCA plot begins to show a pattern, indicating regions where alloy compositions with low and high SFE are absent. Figures 6a and 6b demonstrate that when PC1 (Principal Component 1) is less than -0.2, most of the observations fall within the low and medium SFE classes, with only two outliers in the high SFE class. Similarly, when PC1 is greater than 0.1, most observations belong to the medium and high SFE classes, with two outliers in the high SFE class. These outliers could be due to uncertainties in SFE calculations. Despite these uncertainties, the PCA transformation provides a useful tool for visualizing high-dimensional data, offering insights into the relationships between alloy compositions and SFE values.

3 MACHINE LEARNING MODELS

In this study, seven machine learning models are used to classify the SFE regime (Low, Medium, and High) of steel alloys based on their chemical composition. The workflow involved systematic model development, hyperparameter tuning, cross-validation, and performance comparison. The dataset was split into 80% training and 20% test subsets using random sampling. Models were trained on the training set, and their performance was evaluated on the test set. A 5-fold cross-validation was utilized for hyperparameter tuning and performance validation across all models. A fixed random seed ensured reproducibility of data splits and shuffling procedures. Hyperparameter tuning was conducted using grid search, which systematically explored predefined combinations of parameters to identify those that maximized classification performance. For each model, the optimal hyperparameters and corresponding validation metrics are reported.

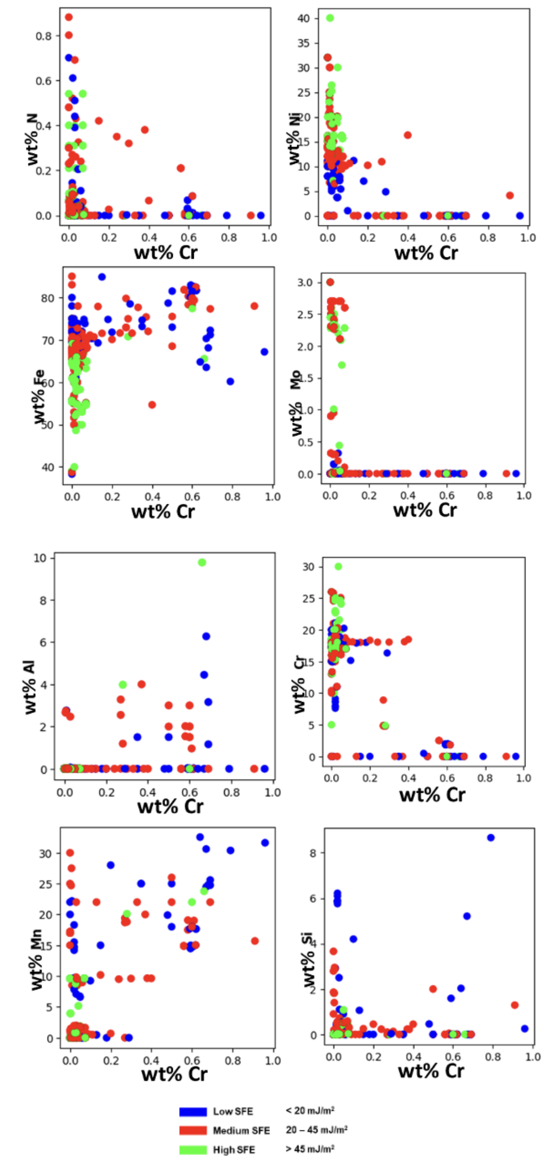


Figure 4: Variation of SFE with wt% Cr for different alloying elements.

3.1 Model Evaluation Metrics

Accuracy was utilized as the primary metric for evaluating model performance. Accuracy is defined as the ratio of correctly predicted instances to the total number of predictions. This provides a straightforward and interpretable measure of overall model correctness and is widely used for initial benchmarking of classification algorithms. It was computed for both the training and test datasets to assess the model's generalization performance and identify signs of potential overfitting or underfitting. While accuracy provides a high-level view of performance, it does not reveal class-specific prediction behavior. For example, in multi-class classification tasks

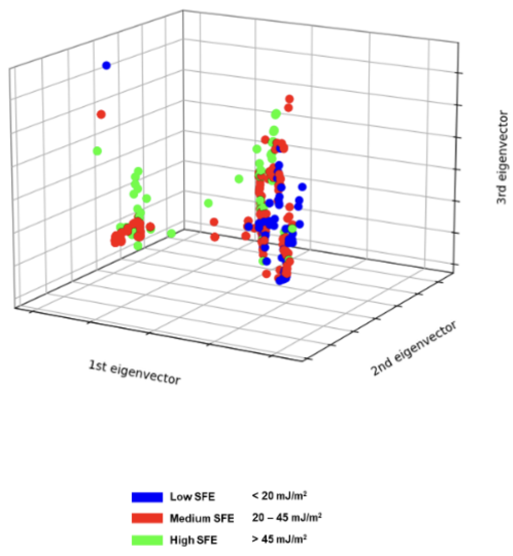


Figure 5: 3D PCA of SFE dataset.

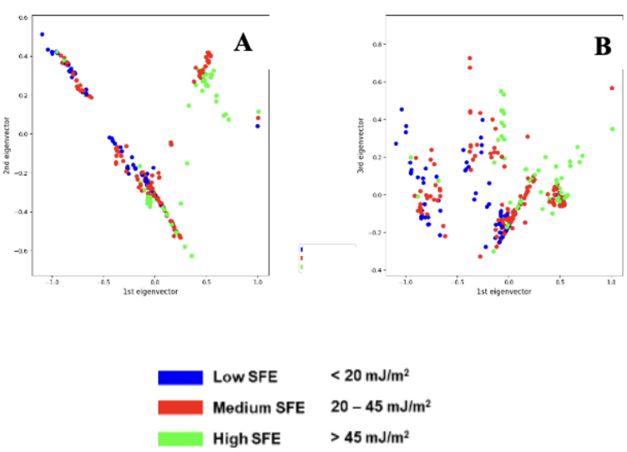


Figure 6: PCA of SFE dataset in 2D with top view (A) and front view (B).

such as predicting Low, Medium, or High SFE regimes, high accuracy can sometimes mask poor performance on minority classes. This limitation is particularly relevant when class distributions are imbalanced or when certain misclassification errors carry greater scientific or practical implications. In this context, further analysis using confusion matrices was conducted to diagnose class-specific prediction errors. The confusion matrix offers insight into how frequently the model misclassifies samples between different SFE categories, which is particularly important in understanding the reliability of predictions across the full range of SFE regimes. This class-level diagnostic can help to identify whether misclassification patterns were random or systematic, and whether specific boundaries were more prone to error. Altogether, accuracy and confusion matrix analysis support a more nuanced view of overall and class-specific model performance, allowing for a more informed comparison between different machine learning models.

3.2 Feed Forward Neural Network (FFNN)

FFNN is one of the simplest classes of artificial neural networks developed, whereby information flows in the forward direction from the input nodes through several hidden nodes, and to a final layer of output nodes. The neurons are interconnected by weights, which form probability-weighted associations between input and output layers through backpropagation. FFNNs can capture complex, nonlinear relationships and elemental feature interactions intrinsic to SFE classification without extra feature engineering. Moreover, they possess a strong universal function-approximation capability.

Data preprocessing for building the FFNN model involves data standardization to ensure all features have a mean of 0 and a standard deviation of 1, leading to more stable training. The model architecture consisted of the following layers:

- Input layer: A dense layer with 64 nodes and ReLU activation, with the number of input dimensions matching the feature count (9 input variables) in the training dataset.
- Hidden layer: Another dense layer with 32 nodes, also using ReLU activation.
- Output layer: A dense layer with 3 nodes, employing softmax activation. 3 nodes were chosen because this is a multiclassification task with three possible outcomes (Low, Medium, High).

The model was compiled using the specified optimizer (from hyperparameter search), a sparse categorical cross-entropy, and using accuracy as a metric for model performance. The hyperparameter selection involved an optimization loop (grid search), iterating over different combinations of:

- Optimizers: Adam and RMSprop.
- Batch sizes: 16 and 32.
- Epochs: 10 and 20.

A stratified 5-fold cross-validation was used to ensure each fold's class distribution closely matched that of the whole dataset. This is useful in classification tasks since there's an imbalance in the distribution of SFE classes. Dropout layers were not added due to the small dataset as this could lead to underfitting, specifically when considering the limited amount of data points in each SFE class.

For each combination of hyperparameters, the dataset is divided into 5 stratified folds, and then a model is created, trained on the training folds, and evaluated on the validation folds. The best model hyperparameters are updated if the current combination yields higher accuracy. The best model, if found, is evaluated on the test dataset to determine its performance on unseen data. The hyperparameter selection iteration selected the rmsprop optimizer, batch size of 16, and 20 epochs. The FFNN model was evaluated to assess its performance on both the training and test datasets. The model achieved an accuracy score of 84% on the training set and 73% on the test set, suggesting some overfitting, where it performs less effectively on the unseen test data. A confusion matrix was generated to gain deeper insight into the model's predictive capabilities.

Table 1: PCA loading scores for the three principal components

PC	C	N	Ni	Al	Fe	Mo	Mn	Si	Cr
1	-0.32609088	-0.01520053	0.30024152	0.0758473	-0.20951403	0.61996516	-0.45110012	0.05278928	0.40373872
2	0.26705977	-0.08920191	-0.24654701	0.06424933	0.20177098	0.77477577	0.2646061	0.00906337	-0.3804223
3	0.02332149	0.62694074	0.02404889	0.04225843	-0.53491664	0.11385471	0.51021697	0.12927422	0.16704913

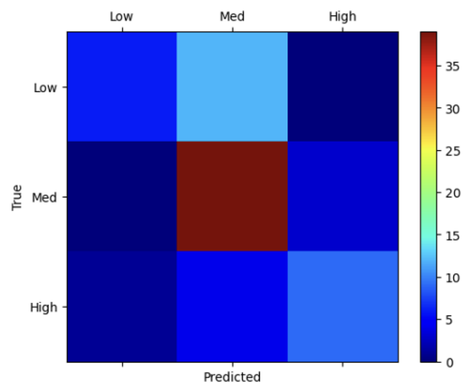


Figure 7: FFNN confusion matrix of validation set.

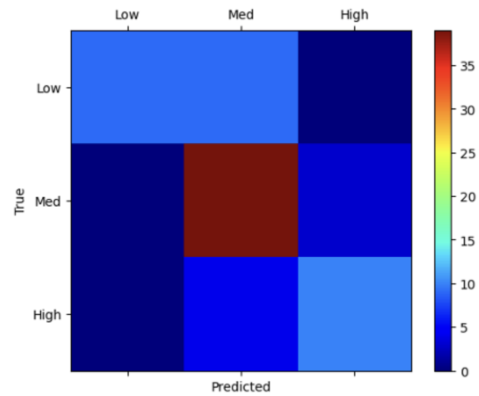


Figure 8: KNN confusion matrix of validation set.

As seen in the confusion matrix in Figure 7, the most amount of misclassification occurs between low and medium classes, with a total of 11 misclassifications here. Some misclassifications also occur between medium and high classes (4 misclassifications). Overall, the model performs the best in predicting the medium class, and this can be attributed to there being more data points for the medium SFE class in the training dataset.

3.3 K-Nearest Neighbors (KNN)

The KNN algorithm is a simple, non-parametric, instance-based learning method used for classification and regression tasks. In classification, KNN assigns a label to a new data point based on the majority class among its k closest neighbors in the feature space, where closeness is typically measured using Euclidean distance. Because KNN relies directly on the training data for making predictions, it does not require an explicit training phase. KNN is particularly useful for classifying SFE regimes because it captures local patterns in feature space without assuming an underlying functional relationship between composition and SFE class by majority vote. This approach allows the classification to reflect compositional similarity to known examples, which can be advantageous in complex alloy systems where SFE trends are not strictly linear. However, the effectiveness of KNN in this application depends on the appropriate choice of k and the feature scaling method, as the algorithm is sensitive to variations in feature magnitude and density of training points in the vicinity of decision boundaries.

A similar approach to data preprocessing for the FFNN was also implemented in the KNN. The KNN is a simple algorithm that does not have a traditional network architecture like other neural networks. The hyperparameter selection involved an optimization loop (grid search), iterating over different combinations of:

- Number of k neighbors: 3, 5, 7
- Weight function: uniform and distance

Following this iteration, the optimal model was found to be 7 number of k neighbors and a distance weight function. The model achieved an accuracy score of 96% on the training set and 78% on the test set. Similar to the FFNN, the most misclassifications occurred between the low and medium classes, with a total of 9, as shown in Figure 8.

3.4 Gradient Boosting Regressor (GBR), CatBoost Regressor (CAT), and Adaptive Boost Regressor (ADB)

GBR, CAT, and ADB regressor methods build predictive models by combining multiple weak learners (typically decision trees) in a sequential manner to improve overall accuracy. The GBR operates by iteratively minimizing a loss function using gradient descent, allowing it to capture complex nonlinear relationships between alloying elements and SFE. The CAT, an optimized variant of GBR, introduces efficient handling of categorical variables and robust regularization, which helps reduce overfitting and improves generalization, important for datasets with moderate size and subtle trends like in this case. ADB, on the other hand, emphasizes correcting errors made by previous learners by re-weighting samples, making it effective in refining predictions where initial models underperform. By applying these boosting techniques, the analysis aimed to leverage their high variance-reduction capacity and ability to model intricate feature interactions, which are critical for understanding how small compositional changes influence SFE across different alloy systems.

The hyperparameters tuned for training the GBR were the number of boosting iterations (100, 200, and 300), learning rate (0.01, 0.1, and 0.2), maximum depth (3, 4, and 5), which represent the depth of individual decision trees within the ensemble, and then the minimum number of samples required to split an internal node (2, 5, and 10). In the CAT model, the hyperparameters tested were the number of boosting iterations (500, 1000, and 1500), learning rate (0.01, 0.05, 0.1), and depth (4, 6, and 8). The ADB model was tested with a number of estimators (50, 100, and 200), learning rates (0.01, 0.1, and 0.2), and max depths (3,4, and 5).

The hyperparameters selected by the grid search are listed below:
GBR:max depth – 3; learning rate – 0.1; min samples split – 2; number of iterations – 100.
CAT: max depth - 4; Iterations – 500; Learning rate – 0.05.
ADB: max depth - 5; Iterations – 200; Learning rate – 0.2.

A stratified 5-fold cross-validation was used to train the GBR and ADB models. However, a simple 5-fold was used in training the CAT model because it does not support stratified KFold directly. All three models performed well on the training dataset by obtaining scores of 92% for the GBR, 92% for the CAT model, and 99% for the ADB. However, the CAT and GBR models performed poorly on the unseen test data by obtaining scores of 60% and 61% respectively. The ADB model performed best on the unseen test data, obtaining a score of 82%.

3.5 Support Vector Machines (SVM)

SVMs are powerful supervised learning algorithms that identify optimal hyperplanes to separate data classes (in classification) or fit a function within a margin of tolerance (in regression). For classification, Support Vector Classifiers (SVCs) aim to maximize the margin between compositional feature clusters corresponding to Low, Medium, and High SFE regimes, making them well-suited for cases with clear interclass boundaries but limited training data. The hyperparameters tuned to optimize the SVM include:

- C range: a range of 10 values logarithmically spaced between 2^{-2} to 2^{10}.
- Gamma range: This influences the decision boundary's flexibility, and a range of 5 values from 2^{-9} to 2^{1} was selected.
- Kernel: The kernel options tested included 'linear,' 'poly,' and 'radial basis function (RBF).'

A stratified 5-fold cross-validation was used on the training set to select the hyperparameters. The iteration process selected a poly kernel, a C value of 64, and a gamma value of 2.0. The SVM model achieved an accuracy score of 90% on the training set and 84% on the test set. The SVM yielded one of the best-performing models, likely due to its robustness in handling high-dimensional feature spaces and non-linear decision boundaries. Given the compositional complexity of steel alloys, SVM's ability to construct optimal separating hyperplanes using kernel functions allowed it to distinguish between SFE classes more effectively than the previous models, which relied on linear separability.

3.6 Random Forest (RF)

Random Forest (RF) is an ensemble learning method that constructs a collection of decision trees during training and outputs either the mode of their classifications (for classification tasks) or the average prediction (for regression tasks). Each tree is trained on a bootstrapped subset of the data, and splits are determined using a

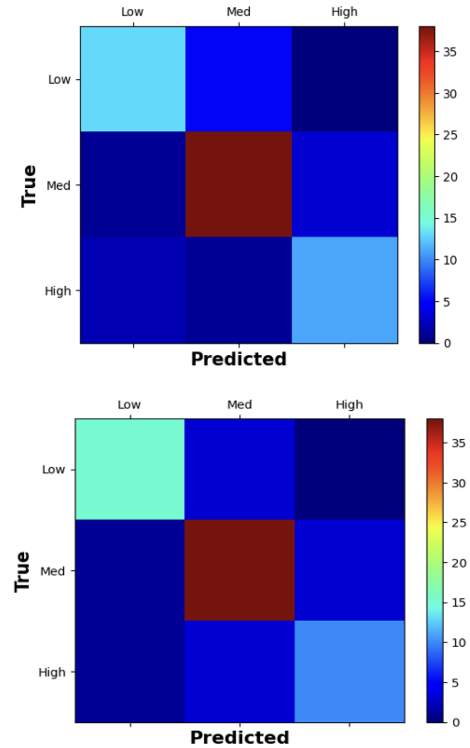


Figure 9: SVM (top) and RF (bottom) confusion matrix with validation set.

random subset of features, which decorrelates individual trees and enhances model generalization. In the context of this SFE analysis, RF was well-suited for capturing complex, nonlinear relationships between alloying elements and stacking fault energy regimes. The model's ability to capture implicit feature interactions and rank feature importance made it particularly effective for multivariate compositional inputs, where some elements can exert a pronounced yet not readily interpretable influence on SFE.

The hyperparameter iteration method was employed to test two key hyperparameters, encompassing the number of trees in the random forest (ranging from 10 to 50) and the maximum number of features considered for node splitting in each tree (ranging from 3 to 5). This resulted in 15 potential sets of hyperparameters to be tested, from which a combination of 40 trees and a maximum feature count of 4 was selected. To determine the optimal hyperparameters, a stratified 5-fold cross-validation was utilized. Impressively, a score of 99% on the training dataset and 85% on the test dataset was achieved by the RF model. The high performance observed for RF in this study likely stems from its ensemble nature, which reduced variance without increasing bias, enabling accurate classification even in the presence of noise or overlapping feature distributions.

Overall, in this study, the confusion matrix derived from the SVM and RF showed the lowest number of misclassifications, as seen in Figure 9, with a total of 5 for the SVM and 3 for the RF. A summary of all model scores is provided in Table 2. The results show that the RF model performed best in both the training and test

Table 2: Summary of model performance across all machine learning models

Model	Model Performance	
	Training data (%)	Testing data (%)
FFNN	84	73
KNN	96	78
SVM	90	84
RF	99	85
GBR	92	61
CAT	92	60
ADB	99	82

data among all the models investigated in this study. Similar better performance is also seen in the confusion matrix for the RF model as well. However, for the next stage of analysis to train the SFE classification model, the RF and SVM models are chosen for further analysis since they both performed best on the unseen test data and had the lowest number of misclassifications in the confusion matrix. Further analysis using the two top-performing models was done to investigate whether one model performed better than the other for the SFE classification tasks.

4 IMPROVING CONFIDENCE IN PREDICTIONS FOR SVM AND RF MODELS

Two machine learning models, SVM and RF, were selected based on their performance on the unseen test data for further analysis and development of a model for the SFE classification task. To reduce misclassifications, there is a requirement for huge amounts of evidence for the particular class the model predicts. To achieve this, a threshold probability is set such that when the model is unsure of an SFE class, it indicates it as fuzzy zones between two classes (low and medium or medium and high). Different threshold probabilities were tested, and after analysis, thresholds of 0.55 for the SVM model and 0.66 for the RF model were selected. These values were chosen because they provide an optimal balance between predictive accuracy and the proportion of fuzzy outputs. As the threshold probability increases, the number of fuzzy outputs also rises. For the SVM model, a threshold of 0.55 was selected, resulting in approximately 10% of the input being classified as fuzzy, with a predictive accuracy of around 88%. Similarly, the RF model's threshold of 0.66 classifies about 10% of the input as fuzzy, achieving a predictive accuracy of about 90%. Tables 3 and 4 provide a summary of the threshold probabilities and their outputs for the SVM and RF models, respectively.

Finally, the model now outputs five SFE classes instead of the original three (low, medium, and high). It introduces two additional classes: Fuzzy-LowMedium and Fuzzy-MediumHigh. For example, the Fuzzy-LowMedium class indicates an SFE value that could potentially fall between the low and medium categories.

5 FEATURE IMPORTANCE ANALYSIS

Austenitic stainless steels are primarily Fe-Cr-Ni alloys with low carbon content. These alloys may also include small amounts of other elements, such as Mn, N, Si, Mo, Al, Ti, and Nb, to impact

Table 3: Threshold probability based on the SVM model

Threshold probability	Predictive accuracy (%)	Fuzzy outputs (%)
0.5	86.49	0
0.55	87.84	9.46
0.66	90.54	22.97
0.7	91.89	38.37
0.8	91.89	51.35
0.9	94.59	82.43

Table 4: Threshold probability based on the RF model

Threshold probability	Predictive accuracy (%)	Fuzzy outputs (%)
0.5	85.14	0
0.6	85.14	1.35
0.66	89.19	9.46
0.7	90.54	14.86
0.8	90.54	25.68
0.9	95.95	50.0

specific properties that can alter the stacking fault energy (SFE) in a non-linear manner. Literature reports indicate that Ni and Fe have a moderate monotonic relationship with SFE [1, 12], while relationships between other elements can be complex. An example of this complex relationship is seen in work done by Vitos et al., where they reported that for Fe-Cr-Ni alloys with 14-16 % Ni, increasing Cr content decreases SFE, whereas for alloys with 17-19 % Cr, increasing Cr content increases SFE [11]. Furthermore, adding Mn to a Fe-Cr-Ni alloy decreases SFE at 0 K, but increases SFE at room temperature when Ni content exceeds 16 %.

This overall non-linearity of SFE values with respect to composition is evident, and therefore, it is important to gain insights into which elements are the key drivers of the model's predictions in this study. To achieve this, a feature importance plot is generated from the RF model, as seen in Fig. 10, which shows the rank of the elements according to how influential they are to the model's predictions. It is seen that Ni has the highest importance in the model prediction, and this finding aligns with what was reported by Wang [12] and Das [4].

SVM models do not inherently provide feature importance in the way the RF models do. Hence, the feature importance plot is only provided for the RF model.

6 VALIDATION OF IMPROVED MODELS

Random austenitic steel alloys, for which SFE calculations have been performed in literature, were used to test the validity of the improved SVM and RF models. Table 5 compares the predictions from the two models with the experimentally or theoretically determined SFE values for different austenitic alloys found in the literature. It is seen that both models perform very well by correctly predicting the SFE class corresponding to the reported experimental values. The RF model classified alloy 5 with a fuzzy zone between medium and high SFE classes. Notably, the reported experimentally

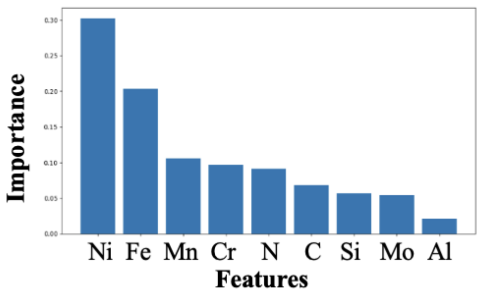


Figure 10: Feature importance plot from RF model.

determined SFE value (45 mJ/m^2 [6]) for this alloy lies directly on the border between the medium (20-45 mJ/m^2) and high SFE (>45 mJ/m^2) class, indicating that the model's prediction reflects the inherent uncertainty near SFE class boundaries rather than returning a misclassification. Similarly, the SVM model also classified alloy 5 as one belonging to a medium SFE class. A comparable case is observed for alloy 2, where the reported SFE value (20 mJ/m^2 [8]) calculated using CALPHAD also lies at the boundary between the low (<20 mJ/m^2) and medium SFE classes, and the two models' predictions differ slightly. Nonetheless, both models are largely in agreement, and neither demonstrates a clear performance advantage over the other. This suggests that using both models in tandem could enhance the reliability of SFE class predictions. An example of using both models in tandem is seen in alloy 3 (45 mJ/m^2 [8]) and alloy 7 (20 mJ/m^2 [13]), which also lie on SFE class boundaries. It is observed that both models agree in their predictions. Therefore, the deformation behavior can be expected to be that observed from the predicted SFE class. Furthermore, for alloys 1, 4, and 6, which are well-defined within SFE classes, it is seen that both models' predictions are accurate and in agreement. Overall, as the SFE dataset grows and more data becomes available for training, each model's accuracy is expected to improve further, potentially reducing the need for a second model as a validation step. In the context of practical applications, both models provide a rapid and efficient screening tool for predicting SFE classes across a wide compositional space, supporting accelerated alloy design. Since the SFE class, rather than the exact SFE value, is more important for predicting secondary deformation behavior, these models offer actionable insights. More importantly, even in compositions where there exists uncertainty in the SFE class, the models do not return outright misclassifications but instead indicate fuzzy zones that can be prioritized for more detailed experimental analysis. Overall, compared to prior approaches that rely solely on thermodynamic calculations or time-consuming experimental methods, both of which can exhibit significant scatter, the machine learning approach presented here provides a robust, reliable, and practical alternative for classifying SFE in austenitic steels.

7 CONCLUSIONS

This study developed machine learning models to classify steel alloys based on their stacking fault energy (SFE) and predict secondary deformation mechanisms. Among the models tested, Support Vector Machines (SVM) and Random Forest (RF) showed the highest accuracy. A fuzzy classification approach, using threshold probabilities, further improved prediction accuracy by accounting for uncertainties near SFE category boundaries. Validation against literature data confirmed strong agreement between the predictions of the models and reported SFE values. These results offer a robust framework for predicting SFE in austenitic steel alloys, aiding in the development of alloys with tailored mechanical properties.

The implementation of machine learning models on platforms like Jupyter Notebook provided an interactive and engaging environment for students. Through hands-on experience with data preprocessing, model training, and evaluation, students gained a deeper understanding of machine learning algorithms and their applications to materials science. The project improved analytical skills and offered practical insights into the potential of machine learning for materials design, enhancing both technical knowledge and problem-solving abilities.

8 STUDENT REFLECTION AND EDUCATIONAL IMPACT

This project provided a meaningful learning experience by applying machine learning to an important materials science problem using an existing SFE dataset. One challenge encountered was understanding the limitations and variability within the dataset, which was compiled from diverse literature sources with varying experimental conditions and methods. Another difficulty was in addressing the non-linear and high-dimensional relationship between alloy composition and SFE by utilizing fuzzy zones to represent regions of uncertainty in predictions. Through this project, the student gained practical skills in data preprocessing, feature selection analysis, dimensionality reduction, data interpretation, implementation, and tuning different machine learning models. For future students undertaking similar projects, we recommend paying close attention to documenting their workflow, including code, parameter choices, and rationale for decisions, to make their work more reproducible and easier to troubleshoot. We also advise students to critically evaluate model output rather than focusing mainly on accuracy, and to make good use of the wealth of available data preprocessing, interpretation, and visualization tools to communicate their results. Furthermore, we advise students to incorporate additional evaluation metrics such as precision, recall, or F1-scores into such complex classification tasks, as this will provide a more complete understanding, especially near class decision boundaries.

ACKNOWLEDGEMENTS

K.N. is partially supported by the NSF grant OAC-2118061.

REFERENCES

[1] Gaurav Arora and Dilpuneet Singh Aidhy. 2020. Machine learning enabled prediction of stacking fault energies in concentrated alloys. *Metals* 10, 8 (2020). https://doi.org/10.3390/met10081072
[2] Jaime A. Castañeda, Oscar A. Zambrano, Germán A. Alcázar, Sara A. Rodríguez, and John J. Coronado. 2021. Stacking Fault Energy Determination in Fe-Mn-Al-C

Table 5: Summary of SVM and RF model prediction compared to SFE values reported in literature

S/N	Alloy composition	Reported SFE (mJ/m^2)	Model Prediction SVM	RF
1	$Fe_{0.03}Mn_{0.02}Si_{0.004}C_{15.6}Ni_{17.5}Cr_{2.5}Mo$	49.6 [14]	H	H
2	$Fe_{16}Cr_{13}Ni$	20 [8]	M	L
3	$Fe_{20}Ni_{25}Cr$	45 [8]	M	M
4	$Fe_{15.9}Cr_{12.5}Ni$	24 [6]	M	M
5	$Fe_{17.8}Cr_{14.1}Ni$	45 [6]	M	Fuzzy M-H
6	$Fe_{18}Cr_{10}Ni_{0.2}N$	23 [13]	M	M
7	$Fe_{18}Cr_{10}Ni_{8}Mn_{0.4}N$	20 [13]	M	M

Austenitic Steels by X-ray Diffraction. *Metals* 11, 11 (2021). https://doi.org/10.3390/met11111701
[3] N. Chaudhary, A. Abu-Odeh, I. Karaman, and R. Arróyave. 2017. A data-driven machine learning approach to predicting stacking faulting energy in austenitic steels. *Journal of Materials Science* 52, 18 (2017). https://doi.org/10.1007/s10853-017-1252-x
[4] Arpan Das. 2016. Revisiting Stacking Fault Energy of Steels. *Metallurgical Transactions A* 47 (2016). https://doi.org/10.1007/BF02646563
[5] Arpan Das, Soumitra Tarafder, and Pravash Chandra Chakraborti. 2011. Estimation of deformation induced martensite in austenitic stainless steels. *Materials Science and Engineering: A* 529 (2011), 9–20. https://doi.org/10.1016/j.msea.2011.08.039
[6] F. Lecroisey and B. Thomas. 1970. On the variation of the intrinsic stacking fault energy with temperature in Fe-18 Cr-12 Ni alloys. *physica status solidi (a)* 2, 4 (1970). https://doi.org/10.1002/pssa.19700020429
[7] Jun Lu, Lars Hultman, Erik Holmström, Karin H. Antonsson, Mikael Grehk, Wei Li, Levente Vitos, and Ardeshir Golpayegani. 2016. Stacking fault energies in austenitic stainless steels. *Acta Materialia* 111 (2016), 39–46. https://doi.org/10.1016/j.actamat.2016.03.042
[8] A.P. Miodownik. 1978. The calculation of stacking fault energies in Fe-Ni-Cr alloys. *Calphad* 2, 3 (1978), 207–226. https://doi.org/10.1016/0364-5916(78)90010-X
[9] Adriana Estela Pontini and Jorge Daniel Hermida. 1997. X-Ray diffraction measurement of the stacking fault energy reduction induced by hydrogen in an AISI 304 steel. *Scripta Materialia* 37, 11 (1997), 1831–1837. https://doi.org/10.1016/S1359-6462(97)00332-1
[10] Cecil G. Rhodes and Anthony W. Thompson. 1977. The composition dependence of stacking fault energy in austenitic stainless steels. *Metallurgical Transactions A* 8 (1977). https://doi.org/10.1007/BF02646563
[11] L. Vitos, J.-O. Nilsson, and B. Johansson. 2006. Alloying effects on the stacking fault energy in austenitic stainless steels from first-principles theory. *Acta Materialia* 54, 14 (2006), 3821–3826. https://doi.org/10.1016/j.actamat.2006.04.013
[12] Xin Wang and Wei Xiong. 2020. Stacking fault energy prediction for austenitic steels: Thermodynamic modeling vs. machine learning. *Science and Technology of Advanced Materials* 21 (2020). https://doi.org/10.1080/14686996.1808433
[13] I.A. Yakubtsov, A. Ariapour, and D.D. Perovic. 1999. Effect of nitrogen on stacking fault energy of f.c.c. iron-based alloys. *Acta Materialia* 47, 4 (1999), 1271–1279. https://doi.org/10.1016/S1359-6454(98)00419-4
[14] Toshio Yonezawa, Ken Suzuki, Suguru Ooki, and Atsushi Hashimoto. 2013. The effect of chemical composition and heat treatment conditions on stacking fault energy for Fe-Cr-Ni austenitic stainless steel. *Metallurgical and Materials Transactions A* 44 (2013). https://doi.org/10.1007/s11661-013-1943-0

Expanding the CyberAmbassadors Program to Include Mentoring for Emerging CI Careers

Katy Luchini-Colbry
Michigan State University
colbryka@msu.edu

Dirk Colbry
Michigan State University
colbrydi@msu.edu

Julie Rojewski
Michigan State University
rojewsj@msu.edu

ABSTRACT

Advanced computing infrastructure has fostered tremendous growth and innovation across research and practice in STEM (science, technology, engineering, math). Cyberinfrastructure (CI) professionals often collaborate with disciplinary experts who want to leverage computation; in order to contribute effectively to this work CI professionals need both technical and professional skills. There are many formal and informal opportunities for the CI workforce to gain technical skills, and the CyberAmbassadors program (NSF Award #1730137) developed new curriculum to provide CI professionals with opportunities to build their professional skills. More than 19,000 participant trainings have been completed, including almost 900 individuals who have earned a certificate for completing the entire CyberAmbassadors program.

This paper describes initial efforts to expand CyberAmbassadors to include training on culturally-aware mentoring skills, with a focus on fostering professional success in the CI workforce – which is still an evolving profession with no single entry path. The new mentoring curriculum will help CI professionals at all levels develop the self-assessment, planning, and networking skills necessary to build strong mentoring relationships that can help them navigate emerging CI career paths. The mentoring curriculum will build on the communications, teamwork and leadership skills training from the existing CyberAmbassadors program, and will offer specialized practice in key career development activities like offering constructive feedback, fostering a growth mindset, developing a mentoring network, and building transferable skills. The new curriculum will also integrate research about the benefits of culturally-aware mentoring, which seeks to provide broad support for mentees with diverse identities and experiences. Once finalized, the new curriculum will be distributed broadly through a national network of volunteer facilitators who provide trainings for their own campuses, companies and communities.

KEYWORDS

Mentoring, Professional Skills, CI Workforce Development

1 INTRODUCTION

As the career pathways for the cyberinfrastructure (CI) workforce continue to evolve, it has become apparent that CI professionals need a variety of disciplinary, technical, and professional skills to support interdisciplinary collaborations. The CyberAmbassadors program developed curriculum to offer training in communications, teamwork and leadership skills in the context of interdisciplinary work in STEM. More than 19,000 participant trainings had been completed by the end of the 2024-25 academic year, including almost 900 individuals who earned a certificate for completing the entire training program. About 150 volunteer facilitators have been trained to use the CyberAmbassadors curriculum to offer communications, teamwork, and leadership skills trainings for their own campuses, companies, and communities.

The CyberAmbassadors project also identified additional needs for professional development – in particular, a need for new training in culturally-aware mentoring of the CI workforce, with a focus on supporting career development in emerging fields. This type of mentoring is distinct from the apprenticeship-style mentoring commonly used in academic and research settings, which is a successful approach for faculty who are preparing graduate students and postdocs for academic careers. The emerging nature of the CI workforce requires a different approach to mentoring and career development in order to support the unique challenges of excelling in roles that are constantly evolving and preparing for jobs that may not exist yet. This paper describes pilot efforts to expand the CyberAmbassadors curriculum to prepare culturally-aware, career-focused mentors to support the CI workforce.

2 BACKGROUND AND RELATED WORK

The CyberAmbassadors program was originally designed to serve CI professionals and the STEM (science, technology, engineering, math) students, postdoctoral trainees, and research scientists who collaborate with and may ultimately join the CI workforce. The remainder of this section highlights a few of the existing technical training programs and supportive resources for the CI workforce and summarizes results from the CyberAmbassadors program.

2.1 Technical Training Efforts

A number of excellent resources for technical training have emerged to serve the CI community, many through NSF workforce development efforts. Indeed, since 2017 the NSF has funded almost 150 grants with the word "CyberTraining" in the title [50]; a cursory review indicates most focus on technical skills such as parallel programming, quantum computing, Data Science, or learning AI. Many of these efforts are reflected in the ACCESS program [2], which coordinates national efforts to make CI hardware and software more accessible to the research community – along with training and resources ranging from workshops to efforts to match researchers with experts who can help. Other federally-funded efforts to support the CI workforce include the Advanced Cyberinfrastructure

https://doi.org/10.22369/issn.2153-4136/17/1/2

Research and Education Facilitators "Virtual Residency," which is an important annual resource for students and practitioners interested in growing their CI careers [51]. The Carpentries [18] offer technical training in data science, computer programming, and library science to support both domain experts and computational specialists.

2.2 Professional Development Programs

Beyond technical training, there are many professional development opportunities and organizations to support the CI workforce. For example, in addition to technical skills the "Virtual Residency" also provides professional skills training, including activities from the CyberAmbassador curriculum. The Campus Champions offer a national network of support for research computing facilitators, including a curated repository of resources and training materials [11, 61] The Campus Research Computing Consortium (CaRCC) [22] hosts forums where individuals can ask questions and seek support for their technical and professional needs, as well as a variety of interest groups like the RCD Nexus (Research Computing and Data Resource and Career Center) [23]. There are also many professional organizations that support more specialized communities, like the Research Data Access and Preservation Association [57] that serves research librarians, archivists, and data scientists.

2.3 Results of CyberAmbassadors Pilot

The CyberAmbassadors pilot project (NSF Award #1730137) resulted in more than 24 hours of training activities to help build professional skills among the CI workforce [14, 43]. Materials are freely distributed as open educational resources, with a Creative Commons license. A non-credit certificate program was added in 2021 to recognize participants who completed 9+ hours of training across the curriculum, which currently includes nine sessions organized into three themes.

Communications sessions cover interpersonal communication skills for solving complex problems:

- First Contact: Communicating with Purpose
- Let's Talk: Communicating about Problems
- It's Complicated: Communicating about Complexity

Teamwork sessions discuss skills for working more effectively in interdisciplinary teams:

- Teaming Up: Effective Group and Meeting Management
- Speaking Up: Effective Presentation Skills
- Leveling Up: Problem Solving and Decision Making

Leadership sessions explore best practices for ethical, inclusive leadership within the CI workforce:

- Leading the Team: Understanding Style and Personality
- Leading the Change: Equity and Inclusion
- Leading with Principles: Ethics

Dozens of universities, professional associations, and research facilities have hosted CyberAmbassadors trainings; and the curriculum has also been used as part of outreach activities like the Mississippi Coding Academy. Tau Beta Pi, the Engineering Honor Society, adopted the CyberAmbassadors materials as the core curriculum of its Engineering Futures professional development program and has committed to hosting the open-source curriculum in the long term, ensuring that it will continue to be freely available to interested trainers and participants.

Table 1: Summary of Pilot CyberAmbassadors Outcomes

Acad. Year	Total Sessions	Total Participants (In-Person | Remote)	Certificates Earned
2017-18	3	344 (88 | 256)	n/a
2018-19	34	1,456(713 | 743)	n/a
2019-20	36	824 (547 | 277)	n/a
2020-21	45	1,919 (58 | 1,861)	72
2021-22	83	2,150 (720 | 1,430)	147
2022-23	167	4,217 (1,990 | 2,227)	188
2023-24	151	3,947 (1,920 | 2,027)	191
2024-25	168	4,427 (1,909 | 2,518)	285
Totals	687	19,284 (7,945 | 11,339)	883

A key part of the success of the CyberAmbassadors project is the "train the trainers" effort to prepare volunteers to use the curriculum to offer professional skills training for their own campuses and communities [13]. As of summer 2025, nearly 150 facilitators have been trained to use the CyberAmbassadors curriculum materials, allowing the program to reach a global audience. Due to the pandemic, the curriculum was adapted for synchronous, interactive, online delivery and that has remained a popular modality in addition to in-person trainings. Table 1 summarizes the outcomes for the original pilot project (a 3-year grant extended to six years due to the pandemic), and demonstrates the program's sustained success since the end of the funding period.

3 MENTORING FOR CI CAREERS

The rapid evolution of cyberinfrastructure means that effectively mentoring the CI workforce requires a focus on lifelong learning and preparing for emerging careers [7, 15, 16, 25, 27, 59]. In order to help mentees build successful careers in CI, mentors need to understand how to support mentees in building transferable skills for jobs that may not exist yet. As the CI workforce expands, it is also important to strengthen mentors' skills for supporting mentees across disciplines, experiences, and cultures. The remainder of this section describes some existing approaches to training mentors, including several CI workforce development projects, and discusses initial efforts to expand the CyberAmbassadors program to offer more focused training for mentoring the CI workforce.

3.1 Existing Approaches to Mentor Training

Formal and informal mentoring relationships have a variety of benefits, which vary depending on the participants and context. For example, the apprentice model of mentoring is well-studied in research contexts where faculty mentors prepare graduate students and postdocs for academic careers [32, 44, 54–56]. In other contexts, mentors may focus on career exploration by helping undergraduates [9, 10] and graduate/postdoctoral students [26, 38] understand career options and gain confidence in their job search. Mentoring can also help individuals identify and develop "transferable skills" like the ability to solve problems, work in diverse teams, and organize complex projects [35, 47, 63]. All of these approaches to mentoring require foundational professional skills like communicating effectively, discussing expectations, setting goals, identifying areas for technical and non-technical growth, and planning for personal and professional development.

The communication, teamwork, and leadership skills covered in the existing CyberAmbassadors program are valuable for building mentor-mentee relationships. Many federally-funded CI workforce development projects include efforts to match and support mentor-mentee pairs, often in the context of building technical and problem-solving skills. Several of these programs have adopted the CyberAmbassadors training as part of the professional development for their participants (CIREN, NSF Award #2230106; SCIPE: CI PIVOT, NSF Award #2321091; CCMNet NSF Award #2216311). This CyberAmbassadors training can provide mentors with a strong foundation in essential professional skills, but the existing curriculum does not offer specific training to help mentors support the unique career development needs of the CI workforce.

3.2 CyberAmbassadors Mentor Training

Early efforts are underway to expand the CyberAmbassadors program to help train mentors (and mentees) in the CI workforce. The new mentoring curriculum builds on the existing training in communication, teamwork and leadership, as these skills are foundational to strong mentor-mentee relationships. The new curriculum is also informed by ongoing research on the benefits of helping mentors and mentees build skills for understanding and integrating their cultural backgrounds and personal experiences in professional contexts [8, 68]. Culturally-aware mentoring approaches value the unique experiences and backgrounds of both the mentor and mentee, and work to acknowledge and integrate their personal and professional trajectories as part of the mentoring relationship. Mentors who take the time to learn about their mentees' backgrounds and adapt their mentoring approaches accordingly can develop stronger, more effective relationships that help foster mentees' career success [5, 33, 37].

The new CyberAmbassadors mentoring curriculum leverages lessons learned from the work of CIMER [28], which provides extensive training and support for culturally-aware mentoring in research contexts (e.g., faculty mentoring graduate students and postdocs). The new curriculum also draws on expertise from the Michigan State University Graduate Career Development Office [29], which specializes in helping graduate/postdoctoral trainees navigate diverse professional contexts and build the transferable skills necessary for success in emerging careers.

Just as with the original CyberAmbassadors training, constructivist and sociocultural pedagogical approaches are being used to develop the new mentoring curriculum. Constructivism and socioculturism are based in Vygotsky's theory of social constructivism [65, 67], which views context as a critical element of the learning process. Constructivism describes learning as an active process of sense-making [36, 42, 52] while socioculturism emphasizes the role of context and the importance of integrating new information with familiar experiences and ideas [24, 40, 41]. Rooted in these approaches, the CyberAmbassadors training strives to include examples and activities from a variety of contexts (e.g., working in an interdisciplinary team, completing a data analysis task, attending a conference or professional meeting). The goal is to build on familiar contexts in order to make it easier for participants to connect the new skills and information they are learning with their past experiences and knowledge.

Just as the original CyberAmbassadors curriculum has a modular format with multiple examples and activities for different audiences, the new mentoring training will be easily customizable for participants with varied career experiences and goals. Table 2 summarizes the types of learning objectives and activities that are likely to become part of the mentoring curriculum; the final materials will be developed based in part on participant feedback.

4 INITIAL MENTORING PILOT

In July 2025, an initial pilot training was developed to explore mentoring in the context of CI careers. 15 participants were recruited from Michigan State University employees, with most coming from the Information Technology and Research Infrastructure communities on campus. Four competency areas were selected as the focus of this initial mentor training: aligning expectations, fostering independence, communicating about problems, and providing feedback. The 3-hour training was conducted in-person in a conference room on campus. At the end of the training, participants were asked to complete a program evaluation form that collected basic demographics; asked about satisfaction with the overall training and the relevance of the content to participants' typical work responsibilities; and offered an opportunity to self-evaluate their confidence or ability to meet the learning goals of the training. 11 participants completed the evaluation form, and the remainder of this section summarizes their feedback on the pilot training.

4.1 Participant Demographics

Of the 11 respondents, 3 had completed an undergraduate degree, 3 had earned a Master's degree, and the remaining 5 had PhDs. 6 participants reported that their primary field of study or area of expertise was in information technology, research computing, or software development; the remaining 5 participants came from a variety of academic backgrounds: computational astrophysics, plant biology, chemistry and materials science, geography (remote sensing), and student affairs administration. 5 individuals reported that they were relatively early in their current careers (1-5 years of experience), while 3 were mid-career (10-15 years of experience) and 3 were advanced in their careers (20+ years of experience). When asked why they decided to participate in the training, all of the participants indicated that they hoped to develop skills and improve their mentoring abilities.

4.2 Overall Workshop Evaluation

Participants were asked to rate their satisfaction with the structure of the overall workshop using a Likert scale of 1=not at all satisfied; 2=slightly satisfied; 3=moderately satisfied; 4=very satisfied; and 5=extremely satisfied. Table 3 summarizes these data.

Participants were also asked to report their satisfaction with specific topics or activities, and to indicate how relevant each was to their typical daily work. Responses were recorded using a Likert scale of 1=not at all satisfied or relevant through 5=extremely satisfied or relevant: Table 4 summarizes this feedback.

Table 2: Example Learning Objectives and Learning Activities for New Mentoring Curriculum

Participants will be able to define mentoring relationships and describe how the relationship between mentors/mentees changes over time
• Introduction to mentoring relationships
• Discussion of mentor/mentee roles
• Brainstorming relationship-impacting events
Participants will be able to assess their technical and professional skills and areas for growth
• Discuss assessment and training resources
• Brainstorm skill-building opportunities
Participants will be able to develop goals for professional growth and a plan for achieving them
• Explore Individual Development Plans
• Practice goal-setting conversations
Participants will be able to identify and leverage opportunities to grow their professional networks
• Introduce the use of multiple mentors
• Explore the role of professional organizations
Participants will be able to initiate a conversation about balancing personal and professional goals
• Practice identifying and clarifying values/goals
• Role play mentor-mentee conversations
Participants will be able to acknowledge and discuss the impact of culture and experience
• Explore differences in culture and experience
• Practice engaging in respectful conversations
Participants will be able to give and receive constructive feedback
• Exploring self-reflection activities
• Role play mentor-mentee conversations

4.3 Aligning Expectations

The aligning expectations segment of the training started by exploring definitions of mentoring in the context of career development. These included mentoring as a "collaborative effort to meet each others' changing needs, and prepare mentees for success in their chosen career" [45] and mentoring as offering "career guidance, skill development, and psychosocial support that increase mentees' self-efficacy, persistence, and career satisfaction" [55]. Participants were introduced to recent research that indicates effective mentoring can prepare mentees for career success [49, 60, 64], enhance mentees' self-efficacy [3, 12, 17, 39], and improve mentees' career satisfaction and retention [6, 58, 66]. The discussion covered characteristics of successful and failed mentoring relationships [61] and an exploration of the various roles and responsibilities mentors may assume (mentoring, supervising, advising, and/or sponsoring) [46]. Participants also discussed ways to help mentees gain experience and prepare for future career opportunities, and what types of "resume building" opportunities might be helpful for mentees. The training also provided practical tips for developing strong letters of recommendation for mentees, and discussed the types of written agreements and documentation that can help mentors and mentees discuss and align expectations.

Participants were asked to complete a self-evaluation of their confidence or ability before and after the training for each of the learning goals. Table 5 summarizes the average responses for the learning goals in the aligning expectations portion of the training. Participants used a Likert scale where 1=very low; 2=below average; 3=average; 4=above average; 5=very high.

Table 3: Average Satisfaction with Workshop Structure

Evaluation Area	Satisfaction
Structure	4.4
Pacing	4.0
Appropriate content for my level of expertise	4.5
Variety of activities	4.1
Opportunities to practice or apply new skills/info	4.0
Facilitators' ability to engage participants	4.8
Facilitators' ability to answer questions	4.9

4.4 Fostering Independence

In the fostering independence section, participants were introduced to the idea of developing a series of scaffolded projects to help new mentees build skills [21]. The first project is an introductory task that is designed to help the mentee acclimate to their new role and give them an early "win" to build confidence. Good tasks for this type of introductory project are things like downloading and installing key software or tools; reviewing orientation materials and completing onboarding tasks; or setting up the computers, accounts, and communication channels (slack, git, etc.) that they will need to work effectively with their new colleagues. As a rule of thumb, a task that would take the mentor 20-60 minutes to complete is often a good candidate for these introductory tasks – but the mentee should be given a full week to complete the task. That leaves time for overcoming unexpected challenges (e.g., needing to wait for access rights or hardware connections) if needed, and allows mentees to exceed expectations early on if they are able to complete the task sooner.

In addition to the introductory task, mentors were encouraged to assign a second-level task that would take them 1-2 hours. The mentee is given two weeks to complete this secondary task, which could be assigned at the same time as the introductory task if appropriate. Often these are "administrative" tasks like re-running an existing experiment or workflow to check reproducibility (and to learn how things are done in their new workplace); reviewing the unit's website or other documentation and identifying a handful of bugs, typos, or updates that are needed; or completing data entry tasks. The goal for this secondary or administrative task is for the mentee to tackle a more open-ended project that encourages them to explore existing resources and find answers more independently.

Once mentees have successfully completed these first two tasks, mentors are encouraged to assign an independent project that might take them 2-4 weeks to complete – but which the mentee has months to complete. Ideally, these independent projects will result in output that the mentee can share with others and potentially receive external feedback (e.g., a presentation, poster, or publication). This not only helps the mentee to build their resume, but also gives them the opportunity to work more independently and take more ownership of a longer-term project. This type of project is often a "filler" that mentees can work on in between other assigned tasks, which helps ensure that there is always something productive to

Table 4: Average Satisfaction with and Relevance of Specific Training Topics or Activities

Topic or Activity	Satisfaction	Relevance
Workshop Introduction	4.5	4.0
Overview of Mentoring	4.5	4.4
Resumes and Recommendations	4.2	3.8
Exploring Expectations Documents	4.6	4.6
Scaffolded Structuring of Projects	4.6	4.3
Paired Peer Mentoring	4.2	3.8
Defining Problem Types	4.5	4.7
Report Inconsistencies Case Study	3.6	3.7
Impact of Communication Styles	4.4	4.5
Best Practices for Giving Feedback	4.3	4.8

Table 5: Average Before and After Self-Evaluation Responses for Aligning Expectations Learning Goals

Learning Goal	Before	After	Diff
Define effective mentoring and explain why it is important	3.1	4.6	1.5
Identify characteristics of successful mentoring relationships	3.5	4.6	1.1
Work collaboratively with mentees to identify goals and action plans	3.3	4.6	1.3
Describe methods for documenting mentoring plans	2.5	4.4	1.9

make progress on even when the mentor may not be able to provide immediate feedback or new assignments.

In addition to strategies for assigning scaffolded tasks that allow mentees to build skills, participants were introduced to the use of peer mentors and partnered work assignments as another way to help mentees gain confidence and independence [4, 10, 20, 30, 39]. There are a number of approaches that can be successful depending on the circumstances, including matching mentees with similar backgrounds or pairing more senior mentees with less experienced ones. In all cases, having the peer mentors meet regularly (typically on a weekly basis) is important: they should be prepared to discuss current successes and challenges with their peer mentor, and to provide feedback on what their partner shares. The goal is for the peer mentors to become the “first resource” for each other: asking questions, brainstorming ideas, troubleshooting solutions, and determining together whether and when it is appropriate to bring an issue to the mentor. This approach helps mentees build their teamwork and communication skills while developing confidence in working with less direct supervision from their mentor.

Table 6 summarizes participants’ self-evaluations of their confidence and ability to achieve the learning goals for the fostering independence segment of the training. In this Likert scale 1=very low; 2=below average; 3=average; 4=above average; 5=very high.

4.5 Communicating about Problems

This section of the mentor training draws directly from the existing CyberAmbassadors program, specifically the module called “Let’s Talk: Communicating about Problems.” The longer-term goal is for the mentoring curriculum to become part of the CyberAmbassadors certificate program, but for this pilot training it was important for participants to have an introduction to some of the communication skills necessary for strong mentoring relationships. This segment begins with a discussion about different types of problems (ability, motivational, interpersonal) and approaches for resolving them in ways that both solve the problem and maintain the relationship [19]. Participants were also introduced to strategies for responding to strong emotions in the workplace [53] and discussed the role of conversational style [62] in communicating effectively within mentoring relationships.

Table 7 shares participants’ self-evaluations of their confidence and ability to achieve the learning goals for the communicating about problems segment of the training. In this Likert scale 1=very low; 2=below average; 3=average; 4=above average; 5=very high.

4.6 Providing Feedback

The final segment of the pilot training focused on providing feedback to mentees. Participants reviewed best practices for why, when, and how to provide effective feedback [1, 31, 34, 48]. Discussion topics included how to balance positive and negative feedback; the value of providing feedback early and often; and the importance of giving specific feedback that can help mentees understand what led to success or what needs improvement. Specific approaches for providing written and oral feedback were reviewed, like asking mentees to include brief descriptions of key tasks as part of their timesheets or to prepare and present a slide summarizing their accomplishments and challenges as part of regular group meetings.

Table 8 includes participants’ self-evaluations of their confidence and ability to achieve the learning goals for the providing feedback segment of the training. In this Likert scale 1=very low; 2=below average; 3=average; 4=above average; 5=very high.

5 DISCUSSION AND FUTURE WORK

While this initial pilot training was small (15 participants, 11 of whom completed the evaluation), it did confirm some areas of strength and highlight some interesting areas for future work. The facilitators who developed and conducted this pilot training also developed the original CyberAmbassadors curriculum, so it is not surprising that there was high overall satisfaction with the structure, content, and pacing of the training – and with the facilitators’ skills at engaging participants and answering questions.

Satisfaction with the specific topics and activities was also high overall, with one outlier in the “Report Inconsistencies” case study. This activity asked participants to discuss in small groups the following scenario: “Your student has a report due in two days, right before the start of winter break. Everyone is anxious to take a break and your student has been working overtime to compile the information. But when you reviewed the draft this morning, you realized there are significant inconsistencies.” This activity was part of the Communicating about Problems segment, and participants were asked to consider what might be the source of the problem and how they might resolve it. Satisfaction is generally high for this activity when it is presented as part of the Communications training within the original CyberAmbassadors program, but pulling out a subsection of that training for this pilot proved to be less effective. Based on participant comments, it seems that the challenges

Table 6: Average Before and After Self-Evaluation Responses for Fostering Independence Learning Goals

Learning Goal	Before	After	Diff
Describe the value of scaffolded projects in fostering mentees' independence	3.0	4.5	1.5
Describe the value of peer mentoring and paired learning experiences for fostering mentees' independence	3.1	4.5	1.4

Table 7: Average Before and After Self-Evaluation Responses for Communicating about Problems Learning Goals

Learning Goal	Before	After	Diff
Define effective problem solving and effective communication	3.4	4.4	1.0
Identify the characteristics of three common types of problems	2.7	4.6	1.9
Practice different processes for diagnosing and solving problems	2.9	4.4	1.5
Describe the impact of communication style and list factors that can influence individual styles	3.0	4.4	1.4

Table 8: Average Before and After Self-Evaluation Responses for Providing Feedback Learning Goals

Learning Goal	Before	After	Diff
Identify best practices for giving feedback to mentees, including why, when, and how to provide effective feedback	3.1	4.3	1.2
Describe the relative benefits of different oral and written communication methods for giving feedback to mentees	3.1	4.2	1.1
Discuss the value of failure as a learning experience, and identify opportunities for constructive failure	3.3	4.1	0.8

stemmed from limited discussion time and a level of discomfort with the ambiguity of the scenario – meaning that participants were not sure they found the "right" answers. In the longer term, the plan is to offer the Communications and Mentoring trainings separately as part of a larger workshop series, which will likely resolve these concerns as there is more time for discussion during a Communications-focused training.

In looking at participants' evaluation of the relevance of different elements of the training to their typical work, three items were rated as only of average relevance: resumes and recommendations; paired peer mentoring; and the case study. These responses are interesting in that all three items focused more on an "academic" context. All of our participants were employed by Michigan State University and most of them supervised student interns or employees. Knowing this, the facilitators designed these activities to reflect experiences with students on a college campus. However, about half of the participants worked in information technology (a support unit, rather than an academic one) and many had experience in industry. One open-ended evaluation response noted that recommendation letters may be less common in industry, and discussion during the training suggested a broader range of working environments that may impact the relevance of peer mentoring approaches (e.g., having only one mentee, or being a mentor but not the supervisor and thus lacking authority to assign partners). As these training materials are refined, attention will need to be paid to the balance of academic and industry contexts and approaches.

In the aligning expectations segment, participants reported gains in their confidence or ability to meet all of the learning goals. During the training, the most robust discussion centered on the four roles or responsibilities of mentors that were introduced, based on [46]:

- **Mentoring:** a focus on nurturing growth, with the goal of understanding mentees' goals and interests and encouraging their progress
- **Supervising:** acting as "the boss" and telling mentees what they need to do, how to do it, and when it is due
- **Advising:** sharing your expertise and telling mentees what you think they should do, based on your experiences
- **Sponsoring:** advocating for your mentee, creating opportunities, and helping them make connections

The overall sense of participants' conversation was that this framework was helpful for thinking about their varied experiences with mentees. Several questions were raised about authority and how to navigate mentoring relationships when you are not a supervisor or employer, and additional time, examples, and activities may need to be allocated to this topic in future trainings.

Participants' evaluation of the fostering independence segment of the training was interesting, as there was a notable increase in confidence for both learning goals (scaffolded projects and peer mentoring). However, these were also topics that many participants found less relevant to their daily work. Given the small sample, it is not clear what adjustments may be beneficial in this portion of the training so additional data will need to be gathered.

The materials presented in the communicating about problems section of the training were the best-developed, since they were pulled from existing CyberAmbassadors training. Participants reported gains in their confidence and/or ability in all four of the learning goals, but the facilitators' assessment after the training was that this segment did not flow well with the other materials. In the original CyberAmbassadors program, there is a logical progression of skills as participants move from the Communications training into Teamwork and then Leadership. While it is not required that trainings be completed in that order, it is encouraged where possible since many participants find the flow between modules to be helpful. The new mentoring curriculum is envisioned as a fourth module, ideally to be completed after the Leadership training. In that scenario, participants would already have completed the training on communicating about problems before taking the mentoring workshops. For this pilot training, most participants had little experience with the CyberAmbassadors program so the decision was made to incorporate elements of the communications training. In the longer term, however, the goal is to offer these separately.

The final segment of the pilot training focused on providing feedback, and while participants reported gains in their ability and confidence with the learning goals overall these were lower than the other segments. It seems likely that part of this was due to a lack of time during the training, as discussions during earlier segments lasted longer than anticipated so the feedback section was a bit rushed. More data will need to be collected from additional pilots in order to better determine what adjustments might be needed for the feedback portion of the mentor training.

Moving forward, the facilitators plan to continue developing and testing the mentoring curriculum materials with varied audiences, likely including graduate students, industry professionals, and participants in disciplinary conferences. The training will also be adapted for delivery in online, synchronous, discussion-based formats. The final materials will be provided to the network of volunteer facilitators who currently offer CyberAmbassadors training, which should allow the new mentoring curriculum to reach a broad national and international audience.

ACKNOWLEDGMENTS

This work builds on the original CyberAmbassadors curriculum that was developed with support from the National Science Foundation under Grant No. 1730137. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Since the conclusion of the grant, logistical and administrative support for ongoing CyberAmbassadors trainings has been provided by Michigan State University and Tau Beta Pi, the Engineering Honor Society. We are also indebted to the volunteer facilitators who continue to offer trainings using the CyberAmbassadors curriculum, and to the participants in the pilot mentor training described here.

REFERENCES

[1] [n. d.]. Mentor Training for Clinical and Translational Researchers. https://www.macmillanlearning.com/college/us/product/Mentor-Training-for-Clinical-and-Translational-Researchers/p/1464152748 Retrieved 2025-07-25. | Macmillan Learning US.

[2] ACCESS. 2023. Advanced Cyberinfrastructure Coordination Ecosystem: Services and Support. https://access-ci.org/ Retrieved 2024-12-13.

[3] Juan Antonio Amador-Campos, Maribel Peró-Cebollero, Maria Feliu-Torruella, Alba Pérez-González, Cristina Cañete-Massé, Adolfo José Jarne-Esparcia, Xavier Triadó-Ivern, and Joan Guàrdia-Olmos. 2023. Mentoring and Research Self-Efficacy of Doctoral Students: A Psychometric Approach. *Education Sciences* 13, 4 (2023). https://doi.org/10.3390/educsci13040358

[4] M. K. Anderson, R. J. Anderson, L. S. Tenenbaum, E. D. Kuehn, H. K. Brown, S. B. Ramadorai, and D. L. Yourick. 2019. The Benefits of a near-peer mentoring experience on STEM persistence in education and careers: A 2004-2015 study. *Journal of STEM Outreach* 2, 1 (2019), 2019.

[5] K. Atkins, B. M. Dougan, M. S. Dromgold-Sermen, H. Potter, V. Sathy, and A. T. Panter. 2020. "Looking at Myself in the Future": how mentoring shapes scientific identity for STEM students from underrepresented groups. *International Journal of STEM Education* 7, 1 (Aug. 2020) (2020), 42. https://doi.org/10.1186/s40594-020-00242-3

[6] J. Balinda. 2023. The Impact of Mentorship Programs on Career Advancement and Satisfaction. *International Journal of Human Resource* 1, 1 (2023), 2023.

[7] N. Berente, S. Ahalt, J. Bottum, D. Brunson, J. Cutcher-Gershenfeld, J. Howison, J. L. King, H. Neeman, J. Towns, N. Wilkins-Diehr, and S. Winter. 2019. The Professionalization of Cyberinfrastructure Personnel?. In *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning) (New York, NY, USA, July 2019)*. 1–6.

[8] S. Black, A. Byars-Winston, I. Cabrera, and C. Pfund. 2022. Enhancing Research Mentors' Cultural Awareness in STEM: A Mentor Training Intervention. *UI journal* 13 (2022), 1. 2022), 36522.

[9] R. S. Bolton-King. 2022. Student mentoring to enhance graduates' employability potential. *Science & Justice* 62, 6 (Nov. 2022) (2022), 785–794. https://doi.org/10.1016/j.scijus.2022.04.010

[10] H. J. Bonner, K. S. Wong, R. K. Pedwell, and S. L. Rowland. 2019. A short-term peer mentor/mentee activity develops Bachelor of Science students' career management skills. *Mentoring & Tutoring: Partnership in Learning* 27, 5 (Oct. 2019) (2019), 509–530. https://doi.org/10.1080/13611267.2019.1675849

[11] M. Brazil, D. Brunson, A. Culich, L. DeStefano, D. Jennewein, T. Jolley, T. Middelkoop, H. Neeman, L. Rivera, J. Smith, and J. Wernert. 2019. Campus Champions: Building and sustaining a thriving community of practice around research computing and data. In *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning) (New York, NY, USA, July 2019)*. 1–7.

[12] D. Brewster and K. Ashley. 2019. African American Male Students' Perceptions of Self-Efficacy, Student Engagement, and Mentoring and Student Retention. *Alabama Journal of Educational Leadership* 6, 2019 (August 2019), 51–61.

[13] A. Briliyanti, J. Rojewski, D. Colbry, and K. Colbry. 2022. Training the Trainers: Preparing Facilitators to Provide Professional Development for Engineers and Scientists. (Aug. 2022). (2022).

[14] A. Briliyanti, J. Wilson Rojewski, K. Luchini-Colbry, and D. Colbry. 2020. CyberAmbassadors: Results from Pilot Testing a New Professional Skills Curriculum. Practice and Experience in Advanced Research Computing. *New York, NY, USA, July* 2020) (2020), 379–385.

[15] C. Brown, R. Johnson, K. Adams, K. Brandt, A. Kleinsasser, J. Deaton, and T. Middelkoop. 2020. Great Plains CyberTeam: A Regional Mentor Approach to Cyberinfrastructure Workforce Development and Advancement. Practice and Experience in Advanced Research Computing. *New York, NY, USA, July* 2020) (2020), 456–460.

[16] S. D. Brown and R. W. Lent. 2013. *Career Development and Counseling: Putting Theory and Research to Work*. John Wiley & Sons.

[17] V. L. Byrne, A. Y. Williams, H. E. Jardine, and A. E. Donlan. 2022. Academic Peer Mentorship as a Leadership Development Experience: Fostering Leadership Self-Efficacy. *Journal of Leadership Education* 21, 1 (Jan. (2022), 2022. https://doi.org/10.12806/V21/I1/R1).

[18] The Carpentries. [n. d.]. https://carpentries.org/index.html Retrieved 2023-12-14.

[19] B. B. Caza, M. Olekalns, and T. J. Vogus. [n. d.]. *How to Mend a Work Relationship*. Harvard Business Review.

[20] D. Colbry. 2014. iCER Interns: Engaging Undergraduates in High Performance Computing. In *Proceedings of the 2014 Annual Conference on Extreme Science and Engineering Discovery Environment (New York, NY, USA, 2014), 71:5*. 1–71.

[21] D. Colbry and K. Luchini-Colbry. 2014. Scaffolded Structuring of Undergraduate Research Projects. In *Proceedings of the 2014 ASEE National Conference (Indianapolis, IN.*

[22] Campus Research Computing Consortium. [n. d.]. ([n. d.]). carcc.org

[23] Campus Research Computing Consortium. [n. d.]. 2024 RCD Nexus Day. ([n. d.]). https://carcc.org/2024/04/24/2024-rcd-nexus-day-register-now/

[24] J. A. Danish and M. Gresalfi. 2018. Cognitive and sociocultural perspective on learning: Tensions and synergy in the Learning Sciences. *International handbook of the learning sciences* 2018 (2018), 34–43.

[25] D. V. Dugas and D. B. Ormand. 2019. Cyber Infrastructure: Training and Mentoring: A way to engage students in their technology future. In *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning) (NY, NY, USA, July 2019)*. 1–7.

[26] P. English, M. J. de Villiers Scheepers, D. Fleischman, J. Burgess, and G. Crimmins. 2021. Developing professional networks: the missing link to graduate employability. *Education + Training* 63, 4 (Jan. 2021) (2021), 647–661. https://doi.org/10.1108/ET-10-2020-0309

[27] H. Fehring and S. Rodrigues. 2016. *Teaching, Coaching and Mentoring Adult Learners: Lessons for professionalism and partnership*. Routledge.

[28] Center for the Improvement of Mentored Experience in Research. [n. d.]. Advancing the Science and Practice of Mentorship. ([n. d.]). https://cimerproject.org/

[29] Graduate Career Development. 2015. https://grad.msu.edu/gradcareerdevelopment Retrieved 2023-12-16.

[30] E. J. Grant-Vallone and E. A. Ensher. 2000. Effects of Peer Mentoring on Types of Mentor Support, Program Satisfaction and Graduate Student Stress: A Dyadic Perspective. *Journal of College Student Development* 41, 6 (2000), 2000.

[31] Dara M. Hall. 2017. The power of feedback: An indicator of mentor effectiveness during student teaching. https://api.semanticscholar.org/CorpusID:80333474

[32] J. Handelsman, C. Pfund, S. Miller Lauffer, C. Maidl Pribbenow, et al. 2005. Entering mentoring: a seminar to train a new generation of scientists. *Ethics in Science and Engineering National Clearinghouse* 164 (2005).

[33] A. O. Hinton, Z. Vue, C. M. Termini, B. L. Taylor, H. D. Shuler, and M. R. McReynolds. 2020. Mentoring minority trainees. *EMBO reports* 21, 10 (Oct. 2020) (2020), e51269. https://doi.org/10.15252/embr.202051269

[34] P. Hudson and S. Hudson. 2014. Mentor feedback: Models, viewpoints and strategies. In *Proceedings of the 2014 Australian Teacher Education Association (ATEA) Annual Conference. T. Aspland*, Australian Teacher Education (Ed.). Association (ATEA, 111–135.

[35] H. Jang. 2016. Identifying 21st Century STEM Competencies Using Workplace Data. *Journal of Science Education and Technology* 25, 2 (Apr. 2016) (2016), 284–301. https://doi.org/10.1007/s10956-015-9593-1

[36] D. H. Jonassen. 1999. Constructivist learning environments on the web: Engaging students in meaningful learning. HTTP://WWW Retrieved 2003-09-24. MOE. EDU. SG/ITEDUCATION/EDTECH/PAPERS/D1. PDF (1999). THE EDUCATIONAL TECHNOLOGY CONFERENCE AND EXHIBITION, SINGAPORE.

[37] K. Kricorian, M. Seu, D. Lopez, E. Ureta, and O. Equils. 2020. Factors influencing participation of underrepresented students in STEM fields: matched mentors

and mindsets. *International Journal of STEM Education* 7, 1 (Apr. 2020) (2020), 16. https://doi.org/10.1186/s40594-020-00219-2

[38] R. L. Layton, V. S. H. Solberg, A. E. Jahangir, J. D. Hall, C. A. Ponder, K. J. Micoli, and N. L. Vanderford. 2022. Career planning courses increase career readiness of graduate and postdoctoral trainees. *F1000Research* 9, 2022 (February 2022), 1230. https://doi.org/10.12688/f1000research.26025.2

[39] H.-G. Le, S. Sok, and K. Heng. 2024. The benefits of peer mentoring in higher education: findings from a systematic review. *Journal of Learning Development in Higher Education* 31 t. 2024 (September 2024). https://doi.org/10.47408/jldhe.vi31.1159

[40] J. Leach and P. Scott. 2003. Individual and Sociocultural Views of Learning in Science Education. *Science & Education* 12, 1 (Jan. 2003) (2003), 91–113. https://doi.org/10.1023/A:1022665519862

[41] J. L. Lemke. 2001. Articulating communities: Sociocultural perspectives on science education. *Journal of research in science teaching* 38, 3 (2001), 2001.

[42] S. M. M. Loyens and D. Gijbels. 2008. Understanding the effects of constructivist learning environments: introducing a multi-directional approach. *Instructional Science* 36, 5-6 (Sept. 2008) (2008), 351–357. https://doi.org/10.1007/s11251-008-9059-4

[43] K. Luchini-Colbry and D. J.-L. Colbry. 2024. Professional Skills to Support Interdisciplinary Work: Lessons Learned from a Successful Collaboration between Universities, Training Programs. *and Professional Societies. (June* 2024 (2024).

[44] Macmillan Higher Education. [n. d.]. Entering Research: A Facilitator's Manual by Janet L. Branchaw; Christine Pfund; Raelyn Rediske. http://www.macmillanhighered.com/Catalog/product/enteringresearchafacilitatorsmanual-branchaw Retrieved 2015-10-16.

[45] R. McGee. 2016. Biomedical Workforce Diversity: The Context for Mentoring to Develop Talents and Foster Success Within the 'Pipeline.'. *AIDS and Behavior. (Jan* 2016 (2016). https://doi.org/10.1007/S10461-016-1486-7

[46] T. Mcglynn. 2017. *Thinking critically about the ways we help our students.* Small Pond Science.

[47] D. McGunagle and L. Zizka. 2020. Employability skills for 21st-century STEM students: the employers' perspective. Higher Education. *Skills and Work-Based Learning* 10, 3 (Jan. 2020) (2020), 591–606. https://doi.org/10.1108/HESWBL-10-2019-0148

[48] A. G. Mubuuke, I. G. Munabi, S. N. Mbalinda, D. Kateete, R. B. Opoka, R. N. Chalo, and S. Kiguli. 2022. Development of a feedback framework within a mentorship alliance using activity theory. *Afr J Health Prof Educ* 13, 4 (Jan. 2022) (2022), 252–258. https://doi.org/10.7196/ajhpe.2021.v13i4.1291

[49] G. Nabi, A. Walmsley, M. Mir, and S. Osman. 2025. The impact of mentoring in higher education on student career development: a systematic review and research agenda. *Studies in Higher Education* 50, 4 (Apr. 2025) (2025), 739–755. https://doi.org/10.1080/03075079.2024.2354894

[50] National Science Foundation. [n. d.]. NSF and Award Search: Download Awards by Year. https://www.nsf.gov/awardsearch/download.jsp Retrieved 2024-12-13.

[51] Henry Neeman, David Akin, Hussein Al-Azzawi, Kevin L. Brandt, Jamene Brooks Kieffer, Dana Brunson, Dirk Colbry, Sandra Gesing, Anna Klimaszewski-Patterson, Claire Mizumoto, Joy A. Pine-Thomas, Anita Z. Schwartz, Horst Severini, Mohammed Tanash, and Daniel Voss. 2020. Cyberinfrastructure Facilitation Skills Training via the Virtual Residency Program. In *Practice and Experience in Advanced Research Computing 2020: Catch the Wave (PEARC '20)*. 421–428. https://doi.org/10.1145/3311790.3396629

[52] M. O'loughlin. 1992. Rethinking science education: Beyond Piagetian constructivism toward a sociocultural model of teaching and learning. *Journal of research in science teaching* 29, 8 (1992), 1992.

[53] K. Patterson, J. Grenny, R. McMillan, and A. Switzler. 2012. *Crucial Conversations: Tools for Talking When Stakes Are High* (second ed.). McGraww Hill.

[54] C. Pfund, J. L. Branchaw, and J. Handelsman. 2015. *Entering Mentoring.* W. H. Freeman.

[55] C. Pfund, A. Byars-Winston, J. Branchaw, S. Hurtado, and K. Eagan. 2016. Defining Attributes and Metrics of Effective Research Mentoring Relationships. *AIDS and Behavior* 20, 2 (Sept. 2016) (2016), 238–248. https://doi.org/10.1007/s10461-016-1384-z

[56] C. Pfund, C. Maidl Pribbenow, J. Branchaw, S. Miller Lauffer, and J. Handelsman. 2006. The Merits of Training Mentors. *Science* 311, 5760 (Jan. 2006) (2006), 473. https://doi.org/10.1126/science.1123806

[57] Research Data Access and Preservation Association. [n. d.]. Home. https://rdapassociation.org/ Retrieved 2023-12-16.

[58] J. Sandvik, R. E. Saouma, N. Seegert, and C. Stanton. 2021. *Should Workplace Programs Be Voluntary or Mandatory? Evidence from a Field Experiment on Mentorship.* Social Science Research Network.

[59] R. B. Schwartz. 2016. The Career Pathways Movement: A Promising Strategy for Increasing Opportunity and Mobility. *Journal of Social* 72, 4 (2016), 740–759. https://doi.org/10.1111/josi.12192 Issues. (2016.

[60] R. L. Stelter, J. B. Kupersmidt, and K. N. Stump. 2021. Establishing effective STEM mentoring relationships through mentor training. *Annals of the New York Academy of Sciences* 1483, 1 (2021), 2021. https://doi.org/10.1111/nyas.14470

[61] S. E. Straus, M. O. Johnson, C. Marquez, and M. D. Feldman. 2013. Characteristics of Successful and Failed Mentoring Relationships: A Qualitative Study Across Two Academic Health Centers. *Academic medicine : journal of the Association of American Medical Colleges* 88, 1 (Jan. 2013) (2013), 82–89. https://doi.org/10.1097/ACM.0b013e31827647a0

[62] D. Tannen. 2011. *That's Not What I Meant!: How Conversational Style Makes or Breaks Relationships.* Harper Collins.

[63] T. Tuononen, A. Parpala, and S. Lindblom-Ylänne. 2019. Graduates' evaluations of usefulness of university education, and early career success – a longitudinal study of the transition to working life. *Assessment & Evaluation in Higher Education* 44, 4 (May 2019) (2019), 581–595. https://doi.org/10.1080/02602938.2018.1524000

[64] T. Vrabie and C. M. Cretu. 2021. Impact of the Mentoring Relationship on the Development of Talented Students – A Narrative Review. *Journal of Educational Sciences* 22, 2021 (2021), 44–62.

[65] L. Vygotsky. 1978. *Mind in Society: The Development of Higher Psychological Processes.* Harvard University Press.

[66] R. P. Walensky, Y. Kim, Y. Chang, B. C. Porneala, M. N. Bristol, K. Armstrong, and E. G. Campbell. 2018. The impact of active mentorship: results from a survey of faculty in the Department of Medicine at Massachusetts General Hospital. *BMC Medical Education* 18, 1 (May 2018) (2018), 108. https://doi.org/10.1186/s12909-018-1191-5

[67] J. V. Wertsch. 1986. *Culture, Communication, and Cognition: Vygotskian Perspectives.* CUP Archive.

[68] V. Y. Womack, C. V. Wood, S. C. House, S. C. Quinn, S. B. Thomas, R. McGee, and A. Byars-Winston. 2020. Culturally aware mentorship: Lasting impacts of a novel intervention on academic administrators and faculty. *PLOS ONE* 15, 8 (Aug. 2020) (2020), e0236983. https://doi.org/10.1371/journal.pone.0236983

Shaping the Future Workforce: Challenges and Lessons Learned in HPC Education from National Labs and Computing Centers

Patrick Diehl
Los Alamos National Laboratory
diehlk@lanl.gov

Ying Wai Li
Los Alamos National Laboratory

Christoph Junghans
Los Alamos National Laboratory

John K. Holmen
Oak Ridge National Laboratory

Elijah MacCarthy
Oak Ridge National Laboratory

Suzanne Parete-Koon
Oak Ridge National Laboratory

Yun (Helen) He
Lawrence Berkeley National Laboratory

Rebecca Hartman-Baker
Lawrence Berkeley National Laboratory

Charles Lively
Lawrence Berkeley National Laboratory

Kevin Gott
Lawrence Berkeley National Laboratory

Lipi Gupta
Lawrence Berkeley National Laboratory

Kristina Streu
Argonne National Laboratory

Yasaman Ghadar
Argonne National Laboratory

Paige Kinsley
Argonne National Laboratory

Jane Herriman
Lawrence Livermore National Laboratory

Erik W. Draeger
Lawrence Livermore National Laboratory

Victor Eijkhout
Texas Advanced Computing Center

Susan Mehringer
Cornell University Center for Advanced Computing

ABSTRACT

Workforce training at national laboratories and computing centers is essential and typically falls into two categories: foundational training for newcomers and advanced training for experienced users. Foundational topics—such as version control, build systems, and basic HPC usage—are largely transferable across institutions, while cluster-specific training varies due to differences in hardware, job schedulers, and local workflows. Training on emerging technologies is split between hardware-specific content and broadly applicable programming paradigms. To reduce redundancy and increase impact, national labs, computing centers, and vendors are collaborating through initiatives like the HPC Training Working Group to share best practices, co-develop materials, and broaden outreach. These coordinated efforts aim to make HPC training more accessible, scalable, and consistent across the community.

KEYWORDS

Workforce Development, HPC, Education, Training

https://doi.org/10.22369/issn.2153-4136/17/1/3

1 INTRODUCTION

The training of the workforce at national laboratories and computing centers has two facets. First, new employees or students, *e.g.*, for internships or summer schools, need to be trained in basic skills. These trainings include build systems, version control, package managers, and how to use high performance compute resources. Second, after users have reached a baseline level of facility with HPC, they need to be prepared for the future with trainings on new hardware, architectures, programming paradigms, and emerging programming languages. Thus, trainings can be categorized into basic skills and emerging technologies. Most basic trainings are common within national laboratories and computing centers. However, some training on how to use compute clusters is different. For example, different job schedulers may be used or some may provide JupyterHub for Python, R, or Julia. Here, most trainings are performed by the local organizations. However, training resources could be shared. For training in emerging technologies, some trainings are specific to a given institution as the institution may have a specific GPU architecture in their supercomputer while other trainings are generic since the technology is architecture independent. Here, vendors or local institutions hold training or hackathons as examples. Additionally, collaborations between institutions for trainings have been established in the last few years. These trainings reflect the current efforts at national laboratories, compute centers, and vendors. Recently, a group of teaching enthusiasts started the HPC training working group[1] and have been

[1] https://olcf.github.io/hpc-training-wg/

meeting monthly. The goals of the working group are the following: *1)* share best practices, experiences, and ideas; *2)* share upcoming user training and outreach events; *3)* share collaboration opportunities (*e.g.*, on training series); *4)* share training materials (*e.g.*, to avoid duplication); and *5)* explore platforms for sharing with the broader community, like HPC-ED[2] [10] and the ACM SIG HPC Education Chapter[3].

The paper is structured as follows: an overview of related work is included in Section 2. The background of HPC trainings and current efforts at labs and compute centers are presented in Section 3 and Section 4, respectively. Section 5 describes some multi-center collaborations. Challenges and lessons learned are discussed in Section 6. Section 7 concludes the paper.

2 RELATED WORK

The collaborative group discussed in this paper, consisting of national laboratories and large centers, has been valuable for finding solutions to problems, discussing new ideas, and for building collaborative efforts. There are many other active working groups in HPC training, *e.g.*, the ACM SIGHPC Education Chapter, the CaRCC AI Facilitation Materials Working Group[4], the ADAC Training, Outreach, & Workforce Development Working Group [5], the US-RSE Education & Training Working Group [6], and other groups affiliated with a topic, region, or type of resource. In addition, there are also material and event collections, such as ACCESS-CI Events and Training[7] and HPC-ED, a metadata catalog of events and materials[8]. Additionally, the Broadening Participation Initiative[9] [11] of the Exascale Computing Project (ECP) provided internships and mentoring via Sustainable Research Pathways for HPC (SRP-HPC) and enhanced collaboration among institutions via the HPC Workforce Development & Retention Action Group (HPC-WDR).

3 BACKGROUND

3.1 Los Alamos National Laboratory

High-performance computing (HPC) training at Los Alamos National Laboratory (LANL) is conducted under the Institutional Computing (IC) program. A comprehensive set of courses is offered to support users across various levels of experience: *1)* Introductory Training: This includes foundational courses on Git, CMake, Spack [4], Unix, HPC fundamentals, and Kokkos [14]; *2)* Advanced Training: Topics such as job scheduling with Slurm and strategies for efficient data management are covered; and *3)* LANL-Specific Training: These sessions focus on tools and resources tailored to the LANL environment, including Charliecloud [12]; and the use of the open science supercomputer.

Recently, a collaborative training on Introduction to Rust [8] was delivered in partnership with the Pacific Northwest National Laboratory (PNNL). Most training participants are novice HPC users from a range of divisions. Notably, the Kokkos training spans three days, starting with a beginner-level introduction, followed by two days of more advanced content.

[2]https://hpc-ed.github.io/
[3]https://sighpceducation.acm.org/
[4]https://carcc.org/ai-facilitation-materials-working-group/
[5]https://adac.ornl.gov/
[6]https://us-rse.org/wg/education_training/
[7]https://support.access-ci.org/events
[8]https://hpc-ed.github.io/
[9]https://www.exascaleproject.org/hpc-workforce/

3.2 Oak Ridge National Laboratory

High-performance computing (HPC) training at the Oak Ridge National Laboratory is primarily through the Oak Ridge Leadership Computing Facility (OLCF), which is a Department of Energy (DOE) Office of Science user facility. Training topics largely relate to making use of the facility's exascale Frontier system and include topics such as best practices, debugging, profilers, tools, *etc.* Where possible, the OLCF collaborates with other centers, especially where interests and users overlap (e.g., ALCF, LANL, NERSC). Examples of such collaborations include a performance portability training series with other examples found in the OLCF Training Archive [10]. In addition to training events, the facility also hosts hackathons for users to collaborate with staff and vendors on challenges of their choosing, monthly user calls on various topics related to the user experience, and outreach events such as an HPC Crash Course, where participants with little to no background work from Unix basics to hands-on activities on OLCF systems.

3.3 NERSC at Lawrence Berkeley National Laboratory

The National Energy Research Scientific Computing Center (NERSC) is the mission high performance computing facility for the Office of Science in the Department of Energy (DOE SC). NERSC is managed by Lawrence Berkeley National Laboratory and is funded by the DOE SC Advanced Scientific Computing Research Office (ASCR).

NERSC deploys advanced HPC and data systems for more than 11,000 users in 1,000+ projects across a wide range of scientific and computational disciplines. More than 50% of the user base is early career (students / postdoctoral researchers), and 1,000+ new users join NERSC each year.

Training [5] and materials are tailored for a variety of user needs: different user personas, different skill levels, various training categories, and different learning styles; and offered in various styles: presentations, presentations with demos, presentations with hands-on components, hackathons, bootcamps, asynchronous learning with Learning Management Systems (LMS), YouTube recordings with professional captions, short videos, archived training materials (slides, recording, exercises).

NERSC regularly collaborates on training efforts with other HPC centers, especially OLCF, ALCF, and LANL, which has proved highly beneficial to both our staff and users. Collaboration efforts span from basic advertisement to co-developing materials, to co-hosting trainings with the same presentations, and customized hands-on exercises for local systems.

3.4 Lawrence Livermore National Laboratory

HPC training at Lawrence Livermore National Lab (LLNL) is offered by both Livermore Computing (LC) and the HPC Innovation Center (HPCIC).

[10]https://docs.olcf.ornl.gov/training/training_archive.html

As the home of HPC systems and support at LLNL, LC offers trainings that help onboard new LC users and upskill existing LC users. The vast majority of LC trainings are held during the summer, when LC experiences an influx of student interns and thereby new users. For example, "LC Getting Started" is a multi-day workshop introducing HPC, parallel programming, schedulers, GitLab, and Python; shorter workshops target particular internship cohorts. Throughout the year, additional trainings are held as needed to introduce new hardware and tools to all users.

The HPCIC runs an annual online tutorial series featuring LLNL-developed open source projects that span the HPC stack, including performance tools, package management, portability suites, and visualization tools. The HPCIC focuses on outreach, and so this tutorial series targets external (non-LC) users and provides Amazon Web Services (AWS) cloud instances to all attendees. Many of these tutorials serve IDEs in the browser from their cloud instances, such that attendees can log in to a ready environment providing the tutorial stack and materials, simply by navigating to the appropriate web address.

3.5 Argonne National Laboratory

The Argonne Leadership Computing Facility (ALCF) is a U.S. Department of Energy (DOE) Office of Science user facility located at Argonne National Laboratory (ANL). The ALCF provides world-class supercomputing resources and expertise to accelerate scientific breakthroughs across a wide range of disciplines. Supporting thousands of researchers annually through programs such as INCITE [11] and ALCC [12], the facility offers access to cutting edge computing systems like Polaris, a pre-exascale system, and Aurora, an exascale system built for the most demanding computational and data-intensive workloads. In parallel, the ALCF invests in workforce development via a year-round training portfolio designed to help new and experienced users make effective use of these systems. Training spans system-specific onboarding and best practices, hands-on workshops and hackathons with staff and vendor experts, and archived materials to support asynchronous learning, with an emphasis on portability, performance engineering, and scalable workflows for Aurora- and Polaris-class architectures. The ALCF also collaborates with peer centers, including OLCF, NERSC, and LANL, to design and deliver joint training that serves overlapping user communities.

3.6 Cornell Center for Advanced Computing

CAC is home to the Cornell Virtual Workshop (CVW)[13], an online site established in 1995. The CVW includes a wide range of topics relevant to computational research, from programming languages to parallel computing to using large cluster and cloud resources to data science and AI. The materials include quizzes, exercises, a personal notebook, and a large glossary of HPC terms. The CVW has about 250K page views annually. The CVW format has demonstrated sufficient reach and efficacy that it has been incorporated in a number of NSF large-scale infrastructure projects, including the Leadership Class Computing Facility and Jetstream Research Cloud.

CAC also offers in-person and video conferencing training, primarily for Cornell computational researchers. Lecture materials and shorter tutorial materials are available through our YouTube channel. Most recently, CAC has offered two lecture series per year to all Cornell campuses, attracting about 250 registrations per topic.

3.7 Texas Advanced Computing Center

At the Texas Advanced Computing Center (TACC) at the University of Texas at Austin, HPC training is done by the HPC group. (Other groups at TACC provide other trainings.) HPC trainings range in length from a half day, with topics such as CMake or the Lmod module system, to whole-week HPC 'institutes'. Since TACC services a US-wide audience, many trainings are online or hybrid; only the institutes are in-person only. In addition to these training classes held at TACC, HPC group members and others also teach for-credit semester-long classes on the UT main campus. The 'Parallel Computing for Science and Engineers', 'Introduction to Scientific Computing', and several programming classes (targeting C++ and Python) are mainstays of this component.

3.8 Working group

During the 2023 annual Supercomputing conference, staff from three national laboratories met to compare training efforts. These discussions helped expose a variety of commonalities including, e.g., training topics taught, types of events offered, and challenges faced. As a result of these discussions, a monthly meeting was established for HPC centers to share their experiences, practices, ideas, and opportunities for collaboration related to HPC training. Monthly calls have included staff from five national laboratories and three academic HPC centers. Early efforts compared training events across centers to help identify gaps and opportunities to collaborate. Subsequent efforts explored platforms such as HPC-ED and the SIGHPC Education calendar [14] for sharing training material and events. Recent efforts organized cross-center training events. Table 1 summarizes training events and asynchronous materials across centers by category and host institution from 2024 to present. The group has continued to meet monthly and in 2025 formalized themselves as the HPC Training Working Group[15]. Monthly meetings are open to all with new centers welcome to the mix.

4 CURRENT EFFORTS AT LABS AND COMPUTE CENTERS

4.1 Los Alamos National Laboratory

In collaboration with ORNL and NERSC, a Julia training session was organized, utilizing Julia notebooks on Perlmutter for hands-on exercises. Building on the success of this initiative, a second iteration is planned for 2026. A series of training sessions was organized in collaboration with NVIDIA. The program comprised four half-day workshops, each dedicated to a distinct topic: Quantum Computing with CUDA-Q [2], HPC SDK, Warp, and Accelerated Python.

[11] Innovative and Novel Computational Impact on Theory and Experiment
[12] ASCR (Advanced Scientific Computing Research) Leadership Computing Challenge
[13] https://cvw.cac.cornell.edu/
[14] https://sighpceducation.acm.org/events/
[15] https://olcf.github.io/hpc-training-wg/

Table 1: Training Events and Asynchronous Materials by Category and Host Institution, 2024 to Present. A single training event may cover multiple topics

Training Topics	LANL	LLNL	NERSC	ORNL	TACC	ALCF	CAC
Parallel Programming	3	4	6	11	2	3	8
Debugging	–	2	3	1	1	1	1
Optimization	1	–	2	–	–	3	1
Profiling	1	2	3	2	1	2	1
Visualization	1	2	2	1	–	2	4
C++	1	–	–	1	2	–	1
Fortran	-	–	2	1	–	–	1
Python	1	2	3	7	–	2	6
Performance Portability	3	2	4	4	–	2	–
GPU programming	1	–	3	3	2	2	1
Storage	1	–	2	2	–	2	–
Job schedulers	1	4	–	2	–	3	1
Build systems	1	2	–	–	1	2	–
Module systems	–	–	1	–	–	2	–
Spack	1	2	–	–	–	–	–
Popular OSS	-	4	2	2	–	3	–
Center-specific Resources	1	8	4	2	3	3	6
Workflows	1	2	1	2	1	2	–
Containers	2	–	2	5	1	1	–
Quantum	1	–	3	8	–	–	–
AI, ML, and Data Science	3	2	6	10	–	6	2

4.2 Oak Ridge National Laboratory

The Oak Ridge Leadership Computing Facility's (OLCF) training team handles training efforts and events at the ORNL. These efforts range from series that run for several weeks to events that run for a few hours. Notable among those are the Frontier Hackathon that spans a week at the minimum where different users of the exascale system participate with their varying science problems. OLCF staff and Frontier COE members join in as mentors and assist the various science teams with their applications with the ultimate goal of optimizing their applications on Frontier. The OLCF also holds new user trainings which aim to equip new INCITE, ALCC and Director's Discretionary (DD) projects with transitioning and using the centers resources. Periodic series on performance portability, profiling, and GPU programming are also organized, some in collaboration with NERSC and ALCF.

The OLCF also partners with SGCI in organizing hackathons targeting faculties in colleges and universities looking to include HPC content into their curriculum [6]. OLCF also collaborates with Intersect360, AWS, HPE and PSC in holding the Winter Classic Competitions, which introduces students to HPC and provides them with hands-on experience with real HPC systems and applications. Another student-focused training program is the Oak Ridge Cluster Academy (ORCA) which trains participants in HPC system administration.

The OLCF HPC Crash Course, originally developed in 2019 for undergraduate and graduate students with little or no HPC experience, introduces core concepts through modular exercises adaptable to varying coding backgrounds. Foundational programming content was added in 2020, with AI and quantum computing modules introduced in 2024 and 2025. Delivered 25 times to date in both virtual and in-person formats, the course is actively maintained by the OLCF training team. Materials are also available for asynchronous learning via two GitHub repositories: `foundational_hpc_skills`[16] for exercises that do not require specialized hardware and `hands-on-with-odo`[17] for HPC system–dependent modules.

To engage future talent early, OLCF adapted the HPC Crash Course for high school students in the DOE WDTS–funded Next Generation Pathways program, providing a curriculum in coding, HPC, AI, and robotics to support students' mentored research projects at ORNL.

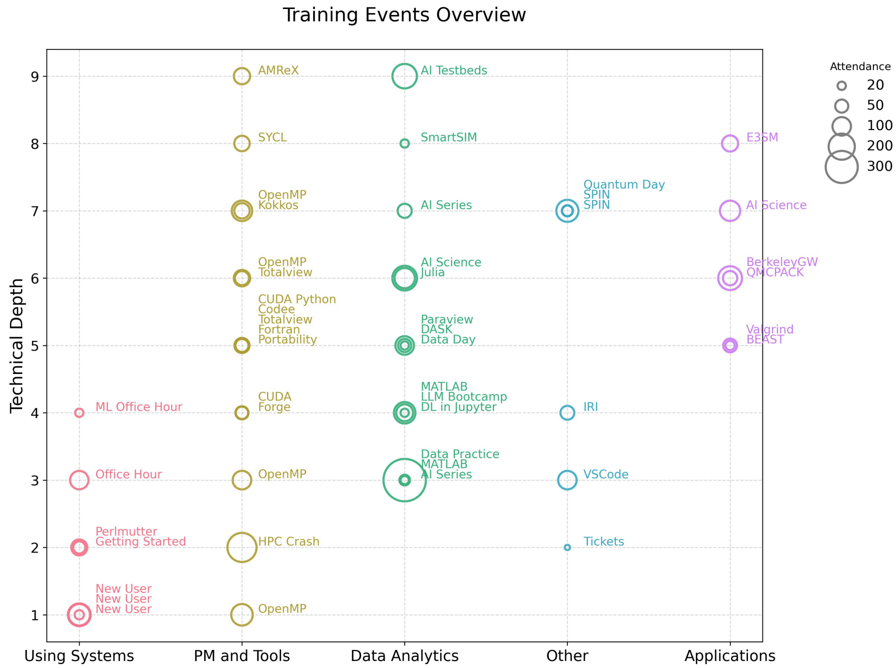
4.3 NERSC at Lawrence Berkeley National Laboratory

Training is a key component of NERSC's strategy of providing scalable services to its 11,000+ users, and NERSC offered a rich set of opportunities for its users to participate in workshops, hackathons, tutorials, outreach, and training events.

Training topics included getting started at NERSC, porting to GPUs, machine learning/deep learning, using performance tools, running applications, programming models, and services. Figure 1 illustrates 2024 training events in five major categories of training topics, organized by level of technical depth and attendance. The above include NERSC hosted/co-hosted training events, and OLCF/ALCF/ECP-hosted training events available to NERSC users.

[16] https://github.com/olcf/foundational_hpc_skills/
[17] https://github.com/olcf/hands-on-with-odo

Figure 1: NERSC Training Events in 2024 with various topics, technical depths, and attendance



4.4 Lawrence Livermore National Laboratory

In recent tutorial improvements, LC has incorporated HPC Carpentry content into its summer tutorials, which has smoothed onboarding for HPC novices.

The HPCIC has been working to improve tutorial deployment in the cloud for our HPC developers. Efforts include wrapper scripts and templates that call the AWS command line interface to ease the provision of AWS cloud instances with minimal configuration or option selection for the developer. We've recently created a Slack bot that is configured to launch instances for the target tutorial on demand during a live event; attendees are given access to the bot so that they can provision their own resources, preventing the hosts from over-provisioning resources on the basis of registrations.

4.5 Argonne National Laboratory

The Argonne Leadership Computing Facility (ALCF) develops and curates a wide range of programs and training materials to support high-performance computing (HPC) training, education, and workforce development. Offerings include in-person and virtual events, as well as on-demand resources such as videos, Jupyter notebooks, presentation slides, and example scripts available through Git repositories and the ALCF On-Demand Training website. [18] ALCF programs are designed for users with varied levels of HPC expertise:

- **Introductory training and education**: Covers foundational topics such as parallel programming with MPI and OpenMP, as well as introductions to advanced concepts like using AI for science.
  - The Intro to High Performance Computing (HPC) Bootcamp is an introductory program hosted by Argonne and designed for undergraduate students with no prior HPC experience. The bootcamp is a unique multi-center collaboration, and more details can be found in Section 5.2.
  - The Intro to AI-driven Science on Supercomputers series is designed to introduce foundational AI concepts to undergraduate and graduate STEM students. [19]
- **Intermediate/Advanced training**: Focuses on performance optimization, debugging, profiling, and emerging technologies such as AI/ML workflows on HPC Systems.
  - ATPESC[20] is Argonne's flagship advanced training program. ATPESC is an intensive two-week program for early-career HPC users, featuring a wide range of HPC topics presented by experts from industry, academia, and national laboratories.
  - The ALCF Developer Sessions are monthly webinars connecting ALCF users to developers of leadership-class systems and software.
- **ALCF-specific training**: Provides guidance on leveraging ALCF systems, tools, and services, including system-specific best practices and job scheduling using PBS.
  - The Getting Started on ALCF Systems trainings are live webinars with demonstrations and hands-on experiences for users new to ALCF systems.
  - The INCITE Hackathon is an annual hybrid event for advanced ALCF users associated with INCITE/ALCC projects

[18] alcf.anl.gov/support-center/training

[19] alcf.anl.gov/alcf-ai-science-training-series

[20] Argonne Training Program for Extreme Scale Computing

or prospective applicants. Over the course of 3 weeks selected teams work closely with ALCF mentors and experts from industry to port, optimize, and scale their applications on current HPC architectures.
- The ALCF Hands-on HPC Workshop is an annual user workshop that introduces ALCF HPC resources to the community. This multi day event combines talks, live demonstrations, and guided hands-on sessions to onboard new research teams and accelerate productive use of ALCF resources.

The ALCF also collaborates extensively with other DOE computing facilities including OLCF, NERSC, and LANL, to co-develop and deliver training materials, workshops, and hackathons, detailed in Section 5. These collaborative efforts address the shared needs of the broader HPC community while fostering knowledge exchange among centers.

4.6 Cornell Center for Advanced Computing

CAC is currently working with the Texas Advanced Computing Center (TACC) Leadership-Class Computing Facility (LCCF), contributing to workforce development via developing online training materials for their resource. CAC is also working on online training material for the Chishiki-AI project[21]. The webinar series offered to all Cornell campuses is currently in the planning stages; input from registration forms and exit surveys, along with registration and attendance numbers, are used to inform the topics offered. YouTube how-to videos on using the local cloud resource are currently being updated, driven by a system update.

5 MULTI-CENTER COLLABORATIONS

5.1 Cross-Center Advertising

Training collaborations include events hosted by multiple centers; we also take advantage of training events from other centers to increase training opportunities for our users. Slack channels, Google Spreadsheets, etc. are used to share our events, which can start in the planning stage. For events hosted by multiple centers, each center creates its own event page. For events hosted by one center and open to other centers, the other centers either make mirror event pages or include the events in their weekly announcements to users.

5.2 Intro to HPC Bootcamp

The Introduction to High-Performance Computing (HPC) Bootcamp is a one-week immersive program designed to introduce STEM students to fundamental HPC concepts and their application to solving complex scientific and energy-related challenges. [22] Established in 2023 as a collaborative effort among national laboratories, including ANL, LBNL, and ORNL, the bootcamp emphasizes hands-on learning, mentorship, and interdisciplinary collaboration [9]. The bootcamp audience is primarily community college and undergraduate students studying domain sciences with an interest in high-performance computing.

[21] https://www.chishiki-ai.org/

[22] intro-hpc-bootcamp.alcf.anl.gov

The first iteration of the bootcamp was funded by the Exascale Computing Project, a joint initiative across DOE national laboratories to develop exascale computing ecosystems, with support from the Sustainable Horizons Institute. The 2025 bootcamp was supported and organized by the Argonne Leadership Computing Facility. In collaboration with LBNL and ORNL, national laboratory staff co-developed the curriculum and led hands-on tutorials. Projects focused on real-world HPC and AI applications reflecting ongoing research across DOE mission areas. The program uses inquiry-based activities to demystify HPC and develop skills in parallel processing, cluster computing, and model building. With a mission to inspire the next generation of computational scientists, the bootcamp promotes lifelong learning and creativity and provides a supportive environment for students from all backgrounds to build a sense of belonging and confidence with high-performance computing and artificial intelligence.

5.3 Julia Training Experience

LANL partnered with ORNL and NERSC to deliver a comprehensive Julia training focused on high-performance computing (HPC). The first day featured two talks highlighting how Julia is being applied in production HPC environments at both ORNL and NERSC. On the second day, participants engaged in a half-day hands-on tutorial that included practical exercises.

To support the tutorial, NERSC granted LANL participants temporary access to the Perlmutter supercomputer, enabling them to run interactive Julia workflows directly through JupyterHub.

5.4 Performance Portability Training Series

The increasing diversity of HPC systems has made code portability more desirable. For example, the major HPC systems at ALCF, NERSC, and OLCF feature Intel-, NVIDIA-, and AMD-based GPUs, respectively. Performance portability layers (*e.g.*, Kokkos) have emerged to help address this diversity. Such layers provide developers with portable abstractions that map to underlying programming models, enabling easy transitions across different architectures. To help educate users about portable solutions, ALCF, NERSC, and OLCF hosted a Performance Portability Training Series [23], which ran from 2023 to 2024. The creation of the series was inspired by the overlap in user bases across centers and included both standalone training events (*e.g.*, RAJA, SYCL) and training series (*e.g.*, OpenMP, HIP).

5.5 Open Hackathons

Hackathons pair selected HPC teams with specific technical experts to achieve targeted computational goals during a multi-day event. HPC hackathons have evolved over the years from in-person dungeon sessions to hybrid multi-week events. The Open Hackathon "series" led by the OpenACC organization[24], is a popular recurring HPC hackathon series hosted around the world, including at academic institutions and DOE science facilities [1].

The OpenHackathon events have proven to be highly impactful to scientific code teams [3, 7, 13], which is attributable in no small part to their collaborative, open design. With OpenHackathon

[23] https://www.olcf.ornl.gov/performance-portability-training-series/

[24] https://www.openhackathons.org

events scheduled approximately once a month across North America alone, the OpenACC team is able to adjust the event design, mentor training, application prioritization, and advertising over time to best suit the rapidly changing needs of the HPC community. Each host contributes to this advancement through a review process held after the event, creating a well-recognized, effective event that is available to the HPC community year-round.

Collaborators to these events share advertising for upcoming hackathons, available mentors, computational resources, where appropriate, and discussions to find solutions to a variety of challenges. OpenHackathon participants include ALCF, TACC, NERSC, Pittsburgh Supercomputing Center, NASA, NCSA, NREL/NOAA/NCAR, Georgia Tech University, Princeton University and many others, who have used the shared expertise to achieve a wide variety of organizational goals.

5.6 Cray User Group Birds of a Feather

In 2024, education and engagement team members from Livermore Computing, the Pawsey Supercomputing Research Center, and NERSC led a collaborative session at the Cray User Group meeting, held in Perth, Australia. The session took place during the Programming Environments, Applications, and Documentation (PEAD) Birds of a Feather (BoF), which includes discussion regarding user-facing issues regarding usage of HPC facilities within the HPE/Cray environment. Because an extension of documentation is training and education (which makes documentation accessible), the session focused on actively developing connections between attendees working on similar educational content. By placing physical sticky notes labeled with different popular and relevant HPC topics onto poster papers around the room, attendees could specify which topics they need help creating content for, which topics they have content for, and which topics they need someone else to make and give them content for. Several connections were made during the session by actively engaging the attendees in this process, instead of asking them to make such a list by themselves, on their own time. The information was disseminated out to each participant, as well as contact information for the various attendees and centers, so ongoing collaborations could be fostered in the future.

6 CHALLENGES AND LESSONS LEARNED

6.1 Resources

6.1.1 Community Resources. HPC centers typically maintain center-specific training material archives and announcement mailing lists. For centers, it can be challenging to advertise relevant events from other centers, as including external events may lead to lengthy announcement emails. For users, it can be challenging to navigate several different training material archives when using HPC systems across various centers. Considering the use of community resources for sharing training material and events could be beneficial for easing the user experience. An example for sharing training material could be making use of platforms like HPC-ED [10].

6.1.2 Computing Resources. Some HPC centers have dedicated testbeds for user training. For centers without such systems, it can be challenging to coordinate user training alongside production user workloads. Similarly, it can be challenging to coordinate user training for new architectures without access to dedicated hardware. Considering collaborating with HPC centers that have access to target hardware could be beneficial for expanding training offerings. Such a collaboration would provide an opportunity to use target hardware before investing in a training system.

6.1.3 Funding Training. Funding requirements for training programs tend to follow a cyclical pattern. In certain fiscal years, significant investment is needed to design and develop new training content. Once the training materials and curriculum are established, funding needs typically decrease, as ongoing costs are limited to delivering the sessions rather than building them from scratch.

A persistent challenge is securing travel funding for external trainers. These trainers — often experts from national laboratories or computing centers — typically volunteer their time and expertise without honoraria. However, they commonly request travel support, which adds to the financial burden.

Another major consideration is the cost of third-party training. While some companies offer limited training free of charge or as part of a maintenance package, many charge substantial fees — particularly for in-person sessions that require multiple instructors on-site. Even virtual trainings can be costly.

One effective strategy for mitigating these expenses is to organize collaborative virtual trainings in partnership with other national labs and compute centers. By sharing both the training and its associated costs, it is easier to justify the expenditure to management and ensure broader access to valuable learning opportunities.

6.1.4 Training Accounts. The combination of attendees from outside the institution participating in a training event and the need to access specific hardware can lead to a problem in handling accounts. TACC previously distributed temporary accounts to attendees who did not already have an account, but this system is now abandoned. Instead, attendees that pre-register are encouraged to create a TACC account, and ‘walk-ins’ can create their account on the spot. For most users, accounts are automatically approved within minutes; someone from user services is at hand for the few cases that need some manual approval, for instance for attendees from certain ‘Countries of Concern’.

At LLNL, LC workshops offer an in-person component where attendees can receive temporary access to existing training accounts and their associated physical tokens for two-factor authentication. This option unfortunately does not work for online participants. LLNL’s HPCIC obviates the need for training accounts on LLNL resources by hosting tutorials on AWS instances, and this approach has allowed us to open attendance to external audiences. Similarly, LLNL employees presenting HPC tutorials at conferences frequently use AWS to provide common computing environments to their attendees.

6.2 Event Planning

6.2.1 Event Format. During the pandemic, virtual meetings became the standard format for HPC training and many other professional events. As some national laboratories continue to support teleworking, offering hybrid training formats has become a logical next step. Interestingly, remote participation remains highly popular—even among onsite employees, who often prefer to attend sessions from their offices rather than in person. This trend highlights the enduring value and convenience of remote training options in the post-pandemic work environment. While the lecture portions of training sessions are generally effective in both virtual and in-person formats, hands-on exercises present greater challenges, particularly when trying to engage both in-person and remote participants simultaneously. One potential solution is to alternate between fully in-person and fully virtual training sessions, rather than using a hybrid format. This approach could improve interaction and support a more cohesive learning experience for all participants.

When a hybrid training format is chosen, for the consideration of accommodating a large number of in-person attendees (such as events targeting summer students) and still opening the event to remote attendees, one practice we have is always asking in-person attendees to join the online session to fully participate online chat and discussions.

6.2.2 Event Outsourcing. A fraction of our training offerings are outsourced to third parties, such as vendors and developers. Hardware and software vendors of our systems have very close relationships with us. They are experts on various training topics of interest to our users, especially in their software usage and performance optimizations. Such training examples include Nvidia CUDA and OpenACC training series, GPU Programming Bootcamps, and HPE Programming Environment and Tools training. We also have contracts with software vendors that could offer training, such as KitWare (Paraview, Pan3D, CMake), Linaro (DDT, MAP), Totalview, etc. Some organizations have expertise in training topics related to general computing and HPC foundation. One such example is Software Carpentry[25] and HPC Carpentry[26].

6.2.3 Event Registration. For collaborative training events, we typically mirror event pages across participating institutions to support local promotion. However, managing multiple registration portals complicates logistics. A centralized registration system would streamline organization by simplifying attendee tracking and enabling consistent data collection for reporting purposes. A key challenge is that some national laboratories use internal registration systems that do not permit access to external participants. To address this, a lab-wide registration infrastructure—similar to eduroam in the academic community—would be highly beneficial, allowing seamless identity verification and access across institutions. One interim solution has been to use Google Forms as a centralized registration point. While this simplifies form submission, it lacks robust user authentication, making it difficult to reliably identify and track participants.

The forms always include questions such as “Which HPC centers do you use?”, and record user names for the purpose of keeping track of attendance from each center. It also helps with adding existing users to compute node reservations using a resource allocation for the training event.

[25]https://software-carpentry.org/
[26]https://www.hpc-carpentry.org/

6.3 Materials and Content

6.3.1 Content Difficulty. Pitching cross-collaborative training content to a variety of participant technical competency levels is challenging. To meet learners where they are, training needs to be able to provide integration across multiple modalities, including project-based learning, peer mentoring, asynchronous materials, and scaffolded tutorials, thus creating a flexible and supportive learning environment accessible to participants with diverse technical backgrounds and lived experiences.

In order to accommodate for different skill levels, trainings can be offered at different levels and in multiple parts. NERSC and other centers have adopted a modular or multi-part structure in their trainings—such as CUDA Part 1, Part 2, and Part 3—so that users can engage at the level appropriate for their background without becoming overwhelmed or disengaged. Pedagogical components have been incorporated into trainings to better help users understand what will be learned. For example, NERSC includes Learning Outcomes for its training events. The list of learning outcomes helps users identify the skills that will be learned in each training, enabling the user to determine whether a training is appropriate for them to join. Additionally, layering content delivery across asynchronous materials, guided tutorials, peer mentoring, and project-based activities has proven useful in supporting different learning styles and pacing.

The rapid pace of change in HPC software standards and programming models creates additional challenges in developing content. Updates to the MPI and OpenMP standards (versions 4 and 5), as well as the continuous evolution of CUDA and other GPU libraries, create moving targets for curriculum development. For example, training a user on CUDA Graphs or OpenMP target offload may be irrelevant if their institution lacks compatible compilers and GPUs. This requires site-specific adaptations or disclaimers that add complexity to shared trainings. To address this, some centers like TACC have started offering dedicated sessions specifically on new MPI standards, while others have integrated cross-center collaborations and hackathons to introduce emerging features in a controlled, supported environment.

6.3.2 Hands-On Content. Hands-on activities during training improves knowledge retention and increases engagement. For training events with a hands-on component, we adapt training materials to specific systems. Users can apply the step-by-step procedures learned in the trainings to their own workflows.

The general concepts of hands-on materials are common to all centers in most cases; only the specific user environments of the centers differ, such as default modules, compilers, batch systems, etc. The task of adapting the hands-on exercises to specific centers is usually straightforward when performed by experienced staff from each center.

6.3.3 Performance Portability. Current and emerging HPC systems are becoming increasingly diverse. For centers, it can be challenging to decide which programming model(s) to train users on. For users, it can be challenging to port to new systems when using vendor-specific programming models. The inclusion of performance-portable programming solutions in training offerings could be beneficial for easing transitions between systems. Such inclusion would help users avoid having to re-write their codes as new systems are introduced.

7 CONCLUSION OUTLOOK

The findings presented underscore the critical role of collaboration between national laboratories and computing centers in shaping the future of HPC workforce development. This collective effort is increasingly important in the context of rapid advancements in artificial intelligence (AI), where HPC serves as a foundational enabler. As AI technologies continue to evolve, the demand for a skilled HPC workforce will grow accordingly, making coordinated initiatives essential for addressing future scientific and technological challenges.

To build on this momentum, the current collaboration will be sustained and strategically expanded. In order to involve additional stakeholders, particularly those national labs and computing centers not yet participating, we plan to organize a Birds of a Feather (BoF) session at either the SC or ISC conference, to serve as a platform to broaden engagement, exchange ideas, and establish new partnerships across the HPC community.

To further enhance the effectiveness of our efforts, we will explore mechanisms for improved organization and sharing of training materials, with the goal of reducing individual workloads and avoiding redundant efforts. Additionally, to strengthen communication with management and program sponsors, we will collaboratively design and implement an effectiveness survey to evaluate and demonstrate the impact of the training initiatives. Let's propel HPC education to infinity, and beyond!

ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Energy through the Los Alamos National Laboratory. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of U.S. Department of Energy (Contract No. 89233218CNA000001). Approved by LANL as LA-UR-25-28001 (Rev. 1). This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231. This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

REFERENCES

[1] Izumi Barker, Mozhgan Kabiri Chimeh, Kevin Gott, Thomas Papatheodore, and Mary P. Thomas. 2023. Approaching Exascale: Best Practices for Training a Diverse Workforce using Hackathons. *Journal of Computational Science* 14, 1 (2023), 17 – 22.

[2] Harun Bayraktar, Ali Charara, David Clark, Saul Cohen, Timothy Costa, Yao-Lung L. Fang, et al. 2023. cuQuantum SDK: A high-performance library for accelerating quantum science. In *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Vol. 1. IEEE, Bellevue, Washington, USA, 1050–1061.

[3] Stu Blair, Carl Albing, Alexander Grund, and Andreas Jocksch. 2015. Accelerating an MPI lattice Boltzmann code using OpenACC. In *Proceedings of the second workshop on accelerator programming using directives*. ACM, Austin, Texas, USA, 1–9.

[4] Todd Gamblin, Matthew LeGendre, Michael R. Collette, Gregory L. Lee, Adam Moody, Bronis R. de Supinski, and Scott Futral. 2015. The Spack package manager: bringing order to HPC software chaos. In *Proceedings of the international conference for high performance computing, networking, storage and analysis*. IEEE, Austin, TX, USA, 1–12.

[5] Yun He and Rebecca Hartman-Baker. 2022. Best Practices for NERSC Training. *Journal of Computational Science Education* 13, 1 (2022), 23–26. http://dx.doi.org/10.22369/issn.2153-4136/13/1/4

[6] John K Holmen, Je'Aime Powell, Alexander Nolte, Elijah MacCarthy, Charlie Dey, Verónica G Vergara Larrea, Suzanne Parete-Koon, and Linda Hayden. 2024. FacultyHack Events: Faculty-Focused Hackathons for High-Performance Computing Curriculum Development. In *Proceedings of the 8th International Conference on Game Jams, Hackathons and Game Creation Events*. ACM, Copenhagen, Denmark, 67–71.

[7] Chip Jackson. 2022. Porting OVERFLOW CFD Code to GPUs: To Hackathons and Beyond!. In *OpenACC and Hackathons 2022 Summit*. OpenACC Organization, Virtual, 1 – 20.

[8] Steve Klabnik and Carol Nichols. 2023. *The Rust programming language*. No Starch Press, San Francisco, California, USA.

[9] Mary Ann Leung, Katharine Cahill, Rebecca Hartman-Baker, Paige Kinsley, Lois Curfman McInnes, Suzanne Parete-Koon, et al. 2024. Intro to HPC Bootcamp: Engaging New Communities Through Energy Justice Projects. *Journal of Computational Science Education* 15, 1 (02 2024), 49 – 56. https://doi.org/10.22369/issn.2153-4136/15/1/10

[10] Susan Mehringer, Katharine Cahill, Charlie Dey, Brian Guilfoos, David Joiner, Richard Knepper, et al. 2025. HPC-ED: Building a Sustainable Community Driven CyberTraining Catalog. *Journal of Computational Science* 16, 1 (2025), 7 – 13.

[11] Suzanne Parete-Koon, Mary Ann Leung, Sreeranjani Ramprakash, and Lois Curfman McInnes. 2023. Exascale Computing Project's Broadening Participation Initiative. *The Journal of Computational Science Education* 14 (July 2023), 53–54. Issue 1. https://doi.org/10.22369/issn.2153-4136/14/1/8

[12] Reid Priedhorsky and Tim Randles. 2017. Charliecloud: unprivileged containers for user-defined software stacks in HPC. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC '17)*. Association for Computing Machinery, New York, NY, USA, Article 36, 10 pages. https://doi.org/10.1145/3126908.3126925

[13] Matthew Smith, Arjen Tamerus, and Phil Hasnip. 2022. Portable acceleration of materials modeling software: CASTEP, GPUs, and OpenACC. *Computing in Science & Engineering* 24, 1 (2022), 46–55.

[14] Christian R. Trott, Damien Lebrun-Grandié, Daniel Arndt, Jan Ciesko, Vinh Dang, Nathan Ellingwood, et al. 2021. Kokkos 3: Programming model extensions for the exascale era. *IEEE Transactions on Parallel and Distributed Systems* 33, 4 (2021), 805–817.

Advancing HPC skills by Developing Large Language Model Retrieval Augmented Generation (LLM-RAG) Systems

Julia Mullen
MIT Lincoln Laboratory
jsm@ll.mit.edu

Sam Corey
MIT Office of Research Computing and Data
secorey@mit.edu

Lauren Milechin
MIT Office of Research Computing and Data
milechin@mit.edu

Riya Tyagi
MIT Office of Research Computing and Data
riyaty@mit.edu

Daniel Burrill
MIT Lincoln Laboratory
daniel.burrill@ll.mit.edu

ABSTRACT

Large Artificial Intelligence (AI) and generative large language models (LLM) are key computational drivers. For researchers developing new tools or incorporating LLMs into their processing pipeline, the scale of data and models require supercomputing resources which can only be met through cloud or High Performance Computing (HPC) architectures. Many of these researchers have deep experience with AI, LLMs, and their research area but are new to HPC concepts, challenges, tools, and practices. To assist this researcher community, the Research Facilitation Teams at MIT Office of Research Computing and Data (ORCD) and the MIT Lincoln Laboratory Supercomputing Center (LLSC) have developed tutorial materials to teach researchers how to build their own Retrieval Augmented Generation (RAG) workflows. Selecting RAG systems as the project focus provides motivation for developing a wide range of skills necessary for efficiently working with LLMs on an HPC system while creating a useful application.

This work details LLM-RAG implementation concerns on two different systems, the design decisions associated with developing the examples, deployment of the workshop training, and the feedback received from the participants. Both the MIT ORCD and MIT LLSC systems are representative of HPC community systems and we plan to refactor the in-person and live virtual workshops into a micro-course built from online, self-paced modules that will be reusable across other HPC centers with slight modifications.

KEYWORDS

High Performance Computing, Artificial Intelligence, Generative AI, LLM, RAG, HPC-LLM Training Modules

1 INTRODUCTION

Large Artificial Intelligence (AI) and generative large language models (LLM) are key computational drivers. For researchers developing new tools or incorporating LLMs into their processing pipeline, the size and scale of data and models require supercomputing resources. The resource needs can be met through cloud architectures or High Performance Computing (HPC) systems. Many of these researchers have deep experience with AI, LLMs, and their research area but are new to HPC concepts, challenges, tools and practices. The paradigm shift from existing tutorials on YouTube and HuggingFace where examples are designed for personal systems to the tasks associated with using a community supercomputing system is massive. As part of this shift researchers need to learn how to install code and packages in user space, select the appropriate compute and memory resources, and run their applications through a scheduler. Additionally, they need to understand the basics of distributed computing in order to understand the challenges associated with running large applications that require multiple nodes for processing.

There are ways to lower these barriers to entry, including the use of containers, modules, well designed teaching examples, and user-guided tutorials. Based on researcher demand, the authors elected to explore approaches to supporting researcher efforts through the development of LLM Retrieval Augmented Generation (LLM-RAG) systems.

A cursory review of in-depth user guides and written tutorials provided by HPC centers highlights two broad approaches for creating LLM-RAG examples that researchers can emulate: provisioning containers [9] and LLM-RAG systems created with a combination of bash and python scripts [3, 5, 10]. In this paper we examine both approaches to building LLM Retrieval Augmented Generation (RAG) systems on an HPC system. The authors represent two distinct supercomputing centers with two distinct approaches to deploying LLM software and training for the research community. Note that when we refer to training in this paper we are referring to educational training.

2 TECHNICAL APPROACH TO EXAMPLE DEVELOPMENT

At a high level an LLM-RAG pipeline executes the following steps:

(1) Load the data set into a machine readable format
(2) Embed the data into a vector database, i.e. vectorstore
(3) Create a workflow that chains the embedding model, vectorstore, and language model together to read user inputs and use them to augment queries and return answers

https://doi.org/10.22369/issn.2153-4136/17/1/4

Table 1: Key Implementation Differences between MIT ORCD and MIT LLLSC

Configuration	MIT ORCD	MIT LLSC
Internet access from compute nodes	Yes	No
Offline model repository	No	Yes
Multi-node examples	No	Yes

The implementation of the LLM-RAG pipeline examples on the MIT ORCD and MIT LLSC systems is impacted by the differences in how these systems are configured and the policies that are applied. The key differences can be seen in Table 1.

In each case, the authors used the Python programming language and the Langchain [1] suite of packages to create the pipeline.

2.1 Deployment via Apptainer

ORCD deployed LLM-RAG using a world-readable Apptainer [11] image containing a prebuilt Anaconda environment to run on our Engaging cluster. We chose container technology over other approaches to minimize the setup time, allowing more class time to be spent interacting directly with the RAG pipeline. To further streamline our deployment, we distribute a shell script to run the pipeline, thus allowing users to interact with the LLM agent using a single command from the command line. The script uses the online ORCD documentation as a knowledge base by default, and with a second shell script users can create an alternative vector store with their own set of documents. The ORCD documentation exists in the form of Markdown files which are generally well suited for vector store creation.

By default, our pipeline uses Mistral AI's 8B Instruct LLM [6]. We chose a small model to reduce download time for our users, limit the GPU resource requirement, and save space on our systems. Furthermore, we found Mistral's models to be the most practical for our use case because they are open source, readily available through HuggingFace, and well-suited for LLM agents. While we explored saving the model in a global location on our cluster to be shared by all users to avoid multiple individual downloads, we found that the download process was a good opportunity for users to learn about storage and bandwidth. Many users might not understand the computational scale of LLMs, and witnessing how many resources are required even for relatively small models should influence their point of view as they develop their own AI workflows.

Our deployment allows for varying levels of engagement among our users. For those who simply want to interact with the LLM agent, they can do so with a single shell command. For users who want to easily customize the pipeline to fit their needs, we added flags that can be used when running the shell scripts, which allow users to set different knowledge bases or change the temperature of the LLM. Advanced users seeking deeper customization, such as enabling multi-GPU inference for larger models or experimenting with different LLM prompt contexts, can treat our code base as a flexible foundation. Its design makes it easy to adapt and extend for more complex use cases.

2.2 Deployment in Offline Mode

To avoid sharing data, queries, and results across the open internet, many organizations discourage or forbid access to public, open LLM tools for professional work. In these environments, researchers require offline models that can be used locally. Downloading models to a laptop quickly becomes infeasible, and researchers often turn to approved cloud providers or organizational clusters. While the use of organizational clusters enables research teams to work from a single system, resource requirements for ever larger models risk overwhelming a shared system. To support the requirements of Lincoln Laboratory projects engaged in LLM research, the LLSC provides a fully on-premise solution. Equiping a useful system for researchers requires providing access to an ever-expanding set of models and a means of serving the models to applications.

The LLSC provides access to open language models through a shared directory on the central file system. Publicly available models used by multiple research groups are downloaded and staged by the LLSC team. To avoid accidental deletion or corruption, the directory is set with read and execute permissions, while only members of the LLSC team have write permission. If a research team needs a model that is not open, they are advised to request a shared (Linux) group and stage the model in the group protected directory.

For models that are small enough to fit on a single node, providing users with the path to the models is enough to get them started. For larger models that require more GPUs than those on a single node, researchers need a way to serve a distributed model from multiple nodes to their application. To address these concerns, the LLSC has chosen to use vLLM [4] because vLLM

- handles communication between nodes,
- coordinates GPU memory through optimized CUDA kernels,
- offers fast model execution with CUDA/HIP graph,
- is well integrated with HuggingFace models, and
- provides an OpenAI compatible API server.

This approach provides enormous flexibility for the researchers allowing them to select a model at run-time by specifiying the model and resources requirements, e.g., number of GPUs, in the submission script. With templated scheduler submission scripts, users only need to change one or two parameters to select a new model for testing. However, one challenge with the LLSC's initial implementation of the vLLM approach was that users needed to develop a fairly deep understanding of Linux, bash scripting, and scheduler (resource manager) use in order to be fully effective. Through multiple iterations of the learning module these challenges have been simplified but not eliminated. The process of training researchers to migrate from more open environments to the on-premise solution is discussed in Section 3.

3 TEACHING AND TRAINING APPROACH

The choice of LLM-RAG provides the training team with the initial opportunity to teach researchers about user space, basic Linux commands, and the shared needs of community systems. The scope of models used in RAG systems motivates the need to better understand how to use GPUs efficiently in a shared community system. For both teams, the LLM-RAG pipeline created the opportunity to build HPC skills using a Just-In-Time approach.

3.1 Learning Objectives

We designed our educational material with two goals in mind: (1) to teach users how to run and customize an LLM agent on a computing cluster and (2) to demonstrate best practices for HPC more broadly through the use of an example application. This results in two sets of learning objectives for students, which we describe below.

3.1.1 LLM-RAG learning objectives. The learning objectives associated with developing and running LLM-RAG pipeline include theoretical concepts, practical implementation concerns and troubleshooting skills. In particular, the teams designed the training so that learners completing the workshops, would

- understand the components of LLM-RAG pipelines, including vector stores, embedding models, and LLM context tuning,
- be familiar with implementing a RAG pipeline using provided code
- be able to modify the RAG pipeline based on individual needs,
- be able to create a vector store based on any set of documents,
- be able to pre-process a set of documents to minimize troubleshooting of vector store creation,
- understand how to launch a vLLM server through an HPC scheduler (MIT LL), and
- be able to use LLMs on a supercomputing/HPC system using GPUs.

3.1.2 HPC learning objectives. While the set of learning objectives associated with running a LLM-RAG pipeline on a supercomputing system were the key drivers of the training, success required that learners develop an understanding about HPC systems and the skills to use them. In practice this meant that the training teams had an opportunity to include a number of general HPC skills in the training. Thus, at the end of the workshop, not only did the learners develop the skills in the previous section, but they also

- understood what containers were and when to use them,
- were able to use containers in an HPC setting (MIT ORCD)
- understood the role of an HPC resource manager (scheduler) (MIT LL),
- were able to submit jobs and monitor jobs through the resource manager (MIT LL),
- gained greater familiarity with Linux command line use,
- had experience monitoring GPU utilization for memory and compute,
- had experience configuring and confirming Python environments in user space,
- had experience configuring user shell environments more broadly, and
- understood best practices for file system use when downloading data and models.

In the next section we discuss how these learning objectives were woven into the course material, through the lecture/presentation and the hands-on practice.

3.2 Workshop Design and Delivery

3.2.1 ORCD workshop. ORCD offered an in-person, two-hour course on running the RAG pipeline as part of the summer HPC course series presented to interested learners in the MIT community. The course was accompanied by a "How-To" guide included in our online documentation. Based on a pre-course survey, learners came from a variety of disciplines at MIT and mostly consisted of graduate students and postdoctoral scholars.

The ORCD team designed our workshop to be highly practical, focusing on hands-on application rather than delving deeply into the theoretical mechanics of LLM agents. That said, we did provide a brief overview of key concepts, including visual explanations of RAG pipeline architecture and the process of embedding documents to build a vector store. This was intended to provide our users with just enough context so that they could make any necessary customizations to the pipeline later on.

The workshop was also highly interactive, with most of the of the teaching taking the form of live coding, which allowed students to follow along on their own. Frequent pauses were taken to ensure that students were not left behind and personal assistance was provided to students as needed. We aimed for a low instructor-to-student ratio, with one teacher and two helpers for every 20 students. Helpers were present to provide one-on-one support to students who had user-specific issues that may not apply to the rest of the class, such as unique environment configurations or storage limitations.

There were a few setup steps that users were required to follow before they could start interacting with the LLM-RAG agent. We informed users of some of these steps in the days leading up to the workshop, but still went over them in person. Users needed to create an account on HuggingFace and generate a user token in order to gain access to the pre-trained LLMs and embedding models used in our setup. Then, from the command line, we went over saving the tokens to individual shell initialization scripts, which we used as an opportunity to teach about shell configuration on shared systems.

We then walked users through requesting necessary resources to run the pipeline. Users connected to our cluster using the shell access to login nodes available through our Open OnDemand web portal [2]. We chose this route over SSH connection to limit the barrier to accessing our systems and ensure a consistent experience among our users, circumventing any individual differences in operating systems and SSH clients. After connecting to a login node, we showed users how to start an interactive job using a GPU. Our Engaging cluster has L40S GPUs, each with 46GB of memory, which is enough to run inference on the Mistral 8B model.

The overall flow of the workshop was intended to start as simple as possible and increase in complexity as time went on. After setup, the first exercise that students performed was running the single shell command that allowed them to interact with the LLM-RAG agent right away. Students initially had to wait for the model to download, during which time we walked through the code and general architecture of the pipeline.

Then, we evaluated the agent by posing different types of prompts. The goal in this portion was to teach the students how to assess the agent's ability to pull information from the knowledge base. In RAG, it is sometimes difficult to know when the agent is generating responses based on parametric memory (i.e., knowledge encoded in the parameters of the LLM) or non-parametric memory (i.e., knowledge gained from external sources of information, such as a vector database of documents). To illustrate this concept, we asked the agent questions that were deliberately absent from the knowledge base (such as, "When did the United States declare its independence?"). The agent would provide a response but would not always elaborate on whether relevant information was present in the provided documents. In the class, we facilitated student input on how to manage this issue.

After experimenting with the LLM-RAG agent on the default ORCD online documentation, we showed students how to create an alternative knowledge base from on another set of documents, and then run the pipeline using the newly created vector store. The example document we used for this portion was a PDF of MIT's Wikipedia page, chosen for its simplicity and relevance. We walked students through uploading files to our computing cluster and running the vector store generation script with necessary flags that point to the new documents. We ran further example prompts using this new agent. The prompts revealed that the agent was less adept at pulling information from the new vector store, suggesting a difference in the embedding model's capacity to encode information from PDFs versus Markdown files.

At the end of the class, students were given the opportunity to try running the LLM-RAG agent using their own documents. Many students came to class prepared with PDFs of academic articles, technical documentation, or other domain-specific sources of information. This portion of the class was highly interactive and collaborative, with many students needing assistance understanding the filesystem on our cluster as well as curating their documents to fit the pipeline. This portion created opportunities to discuss various HPC concepts with students, including memory, GPU utilization, and storage.

3.2.2 LLSC workshop. The LLSC HPC-RAG training module was initially delivered as part of an 8 week virtual professional education course on Large Language Models. Additional offerings included a hybrid short course on the theory and practice of LLMs and a second run of the 8 week virtual LLM course. In each case, the goal of the HPC-RAG module was to provide students with hands-on exposure to LLMs on an HPC system prior to the start of their student-defined, mentor-guided project. We note that the mix of students included many who haven't written code in years and others with deep backgrounds in LLMs but limited experience with HPC systems.

The long courses were designed so that each module, or week, included online asynchronous learning material: videos, articles, and hands-on practice along with a 90 minute live, synchronous session. When designing the RAG module we reviewed the skills and knowledge that students needed to work independently, the information required for setting up their HPC accounts to work through the RAG example, and the skills required to be successful with the example. Based on the review, the asynchronous online preparatory work included

- an introduction to HPC including a short video about the components of HPC systems and how they differ from a laptop,
- information on accessing the HPC system through the web portal including a short question to confirm access,
- information on Jupyter Lab and Jupyter Notebooks including how to start them on the system and how to use them, and
- best practices for Jupyter Notebook use.

Similar to the ORCD workshop, the live, synchronous session included a presentation introducing and explaining RAG pipelines, their architecture, and the hands-on RAG activity. The hands-on portion of the session used live coding extensively within Jupyter Lab running on the HPC system. As with ORCD's workshop, this approach was chosen to ensure access and a consistent environment for all students. To work with the RAG example, students needed to first start a terminal in Jupyter Lab and submit a job to the scheduler to start the vLLM server. Once the server was running, they used the Python scripts provided to set up and run the RAG pipeline. To provide additional assistance for students during the hands-on activity, LLSC staff from the Computational Science and Engineering team joined the session and used breakout rooms to work with students. Student feedback on the module was very good and students suggested that a re-run of the course should include more interactive modules.

While the students were successful during the synchronous session, when they started working on their guided projects, their first challenge was the lack of internet connectivity from the compute nodes. Most researchers are used to using "sudo" or "pip" to install packages as needed, sometimes even as they start their application, so the lack of internet access caused some confusion. The work around is to set up the compute environment from either a login or download node before starting work. Students struggled, though documentation was provided. To address this the training team re-evaluated the learning objectives in preparation for updating the module prior to the second run of the 8 week course.

The "Introduction to LLMs: Theory and Practice" short course provided an ideal opportunity to refactor the initial module and test it with a live class. There were two significant changes in the learning material. The first resulted from re-factoring the way that we addressed the challenges associated with the lack of internet connectivity from the compute nodes. The new version included content about required tasks resulting from the constraint. In particular, the newly created training materials built on an analogy to traditional HPC processing where applications require completing pre-processing, processing, and post-processing tasks. In the context of LLMs or interpreted packages in general, the pre-processing stage (Phase 1) is the creation and confirmation of the user's computing environment. The processing stage aligns with the execution of the LLM pipeline and application. Students in the short course found this separation of the two phases clear and easy to follow. They also understood that this approach would be required anytime there were changes in the environment, such as new or updated packages. This refactoring has the additional benefit of motivating the need to understand Linux environments, how to use modules, and how to troubleshooting environment issues.

The material in Phase 1 required students to use the Linux command line, load module files, install packages in user space, confirm package installation, and test the environment. In some cases, students had to start the Python interpreter at the command line to download or update additional packages. The ability to do all this highlighted their HPC skills and prepared them for their individual projects in addition to the RAG pipeline example.

The second major change was to convert the scripts associated with setting up and running the RAG pipeline into cells within a Jupyter Notebook. This new approach meant that students were still required to use the Linux command line to submit a job to the scheduler to start the vLLM server, but once started they were able to explore the pipeline in a Notebook. One advantage of this approach was the ease with which new cells could be added and used to experiment with a range of prompts. The output from previous cells was still available in the Notebook making it easy to compare the prompts and outputs.

To increase the usability of the RAG example, the second 8 week course was augmented with an extra short lab in Module 3, the week prior to the RAG pipeline (Module 4). The augmented lab required students to complete the Phase 1 steps during the Week 3 asynchronous work. In addition, they were asked to upload documents related to their projects. During the synchronous session for Module 3, students used and modified a Jupyter Notebook to load their documents, select an embedding model and embed their documents in a vector store. This redesign had multiple benefits: (1) because some documents had errors during the embedding it prompted a discussion about data preparation, (2) at the end of the session student documents were embedded in a vector store and ready to use for the RAG example, and (3) we uncovered and resolved issues that would impact the success of the RAG example the following week.

4 LESSONS LEARNED

The MIT ORCD and LLSC teams took away many lessons from the RAG training workshops—some related to provisioning the tools and others related to training.

4.1 Training

The examples created by the MIT ORCD and MIT LLSC teams were developed and tested across multiple training events. Through this process, the authors learned:

- **Providing templates is helpful, but users still need context.**
  The shell scripts administered by ORCD and the Jupyter Notebook administered by LLSC allowed students to run things quickly, but a more thorough understanding of the pipeline is necessary for a student to personalize things to fit their own use case.
- **Providing editable scripts that are very similar to default scripts encourages users to experiment in a scaffolded environment.**
- **Converting demonstration examples to teaching examples requires explaining what the steps are and why they are necessary.**
- **Teaching proper HPC resource utilization cannot be ignored.**
  Students who are new to HPC or LLMs may not have a sense for the resources required to run these models. Many students are eager to start experimenting with larger models, but without understanding their computational cost, they may begin to monopolize resources that they do not need. For example, an idle Jupyter Notebook that has multiple GPUs allocated has a negative impact on other researchers in the community who need to run their own computational tasks.
- **Unique use cases may cause the pipeline to break.**
  It is important to schedule in extra time for debugging individual use cases. In our classes, example documents that some students brought did not perfectly integrate with the pipeline that we had built. Furthermore, for the ORCD course, some students did not have enough home directory storage to download the LLM, so we had to walk them through additional steps to save the files elsewhere.
- **Tailoring material for multiple audiences is complicated.**
  Our pre-workshop survey revealed that students joined the course for various reasons. Some were trying to develop an LLM-RAG pipeline of their own, but others attended to learn about containers, and others simply wanted more experience using an HPC cluster. While this was in accordance with our goal—to teach various concepts through a single example—this made it difficult to choose which aspects of the course to cover in more detail.
- **Low teacher-to-student ratios are vital.**
  Due to the interactive nature of the course, students learned through trial and error. We found it essential that enough instructors were present to provide one-on-one assistance to students.
- **Students are interested in agent effectiveness.**
  Many students sought a practical application of this pipeline to fit their needs as researchers. As a result, they needed to know how accurate the agent is at summarizing information to be reliable for their research. In the future, students would benefit from a more formal teaching section on evaluating the LLM-RAG agent.

4.2 Provisioning Resources

In addition to the lessons learned from teaching, the teams took away a number of lessons associated with provisioning the systems for these workshops:

- **The LLSC team noticed a peculiar usage pattern with vLLM.**
  A number of nodes were running jobs that had very spiky usage, idle for long periods with intermittent workloads. While this behavior is common for LLMs, it is particularly problematic with multi-node models and made worse when users do not shutdown the vLLM server. Various solutions were applied including a time limit on the vLLM server and configuring the vLLM server so that research groups could share a single vLLM server instance.
- **The pipeline should be configured to use optimal storage spaces.**
  The ORCD team's setup involved users downloading models to their home directory, which take up at least 15GB of space. Because not all users have that much available space, we encouraged them to choose alternate locations, such as temporary scratch storage.

- **A vector database may be edited if a different LLM is used.** ORCD's configuration stored its default vector database in a global, read-only location on our computing cluster. This became problematic when users attempted to experiment with alternative LLMs, as the pipeline tries to write to the vector store during execution if the LLM is changed. To resolve this, we modified the pipeline to automatically copy the vector database to the user's home directory, ensuring write permissions and preserving flexibility in model experimentation.

5 SUMMARY AND NEXT STEPS

The authors developed LLM-RAG tutorials and used them to successfully train researchers who were new to using supercomputing systems. These learning modules should be reusable at other HPC centers with slight modifications.

The ORCD team in particular hopes to make our approach more generalizable by utilizing alternative software that helps make the building of LLM agents more streamlined, especially on a shared computing cluster. These tools, such as Ollama, vLLM, or Morphik, are greatly valuable for scaling up workflows and simplifying the development process. Furthermore, ORCD hopes to continue collaboration with MIT undergraduate students, who have been integral to this project so far. We will continue to offer this course to our community, constantly iterating and improving upon our setup to make this a more valuable resource. Our course materials, including our "How-To" guide as well as our GitHub repository, are available public access online [8] [7].

The LLSC team is in the process of converting the individual modules and hands-on exercises to a short "Building LLM pipelines on HPC systems" course that should be easily modifiable for other centers. The goal is to provide learners with a skeleton they can modify and tweak to explore LLM pipelines along with a guide to understand the general approach. As a result, the LLSC approach doesn't use any special tools, beyond vLLM, and should be transferrable to other HPC systems. In addition, the authors would like to create a teaching guide that can be used by other members of the education and training community.

ACKNOWLEDGEMENTS

DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited. This material is based upon work supported by the Department of the Air Force under Air Force Contract No. FA8702-15-D-0001 or FA8702-25-D-B002. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Air Force. JM would like to acknowledge the MIT SuperCloud and LLSC teams for assistance with hosting and deploying the workshop materials and facilitating the training. SC and LM would like to acknowledge the ORCD team, especially Jonathan Murray for helping out during the workshop.

REFERENCES

[1] Harrison Chase. 2022. LangChain. https://github.com/langchain-ai/langchain Release date: 2022-10-17.
[2] Jeremy S. Hudson, Joel Welling, Alan Chalker, Trey Dockendorf, Jeff Ohrstrom, Andreas Dilger, and Boyd Wilson. 2018. Open OnDemand: A web-based client portal for HPC centers. *Journal of Open Source Software* 3, 25 (2018), 622. https://doi.org/10.21105/joss.00622
[3] Owain Kenway. 2023. *Running a Large Language Model on our HPC systems*. Retrieved March 12, 2025 from https://www.youtube.com/watch?v=taUixaGpSbs
[4] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.
[5] Marco Magliulo. 2025. *Inference of Hugging Face's Pre-Trained LLMs on HPC Systems*. Retrieved March 12, 2025 from https://www.youtube.com/watch?v=0YAyy3ACYek
[6] Mistral AI. 2024. Ministral-8B-Instruct-2410. https://huggingface.co/mistralai/Ministral-8B-Instruct-2410. Fine-tuned instruction-following language model with 8B parameters, released under the Mistral AI Research License.
[7] MIT Office of Research Computing and Data. 2025. *orcd-rag*. Retrieved September 9, 2025 from https://github.com/mit-orcd/orcd-rag
[8] MIT Office of Research Computing and Data. 2025. *Running Your Own Retrieval-Augmented Generation (RAG) Model*. Retrieved September 9, 2025 from https://orcd-docs.mit.edu/recipes/rag/
[9] NYU HPC. [n.d.]. *LLM of HPC*. Retrieved March 12, 2025 from https://sites.google.com/nyu.edu/nyu-hpc/training-support/general-hpc-topics/ai-at-hpc-tips/llm-on-hpc
[10] QC University eResearch. [n.d.]. *Large Language Models/Natural Language Processing on HPC System CPU or GPU modules*. Retrieved March 12, 2025 from https://eresearch.cqu.org.au/high-performance-computing/hpc-user-guides-and-faqs/running-llms-on-the-hpc-system/
[11] Singularity Developers. 2021. Singularity. https://doi.org/10.5281/zenodo.1310023

HPC-ED: Testing Automated Agents to Assess the Quality of Training Resource Metadata

Habiba Morsy
Kean University
University of Virginia
morsyh@kean.edu

Essence Toone
Kean University
toonee@kean.edu

Charlie Dey
Texas Advanced Computing Center
charlie@tacc.utexas.edu

Zilu Wang
Cornell University Center for Advanced Computing
zw427@cornell.edu

Mary Thomas
University of California, San Diego
mpthomas@ucsd.edu

David Joiner
Kean University
djoiner@kean.edu

ABSTRACT

We present a proof-of-concept system for automating quality assurance in the HPC-ED federated training catalog using large language models (LLMs). The HPC-ED catalog system integrates metadata crawling, video transcript extraction, and model-based evaluation to score and provide recommendations on metadata quality at scale.

KEYWORDS

HPC, Metadata Quality, Large Language Models

1 INTRODUCTION

1.1 The HPC-ED Catalog

The HPC-ED project collects item-level metadata for training objects in partner catalogs and digital libraries to enhance content discovery for the HPC and broader cyberinfrastructure (CI) communities. Existing projects have produced a wide range of training materials, including tutorials, workshops, course modules, and recorded events. Surveys of HPC training providers and users have consistently identified difficulties in finding resources at the right depth, filtering results, and identifying trusted sources [1].

To address these issues, the **HPC-ED** project was launched to improve discovery, sharing, and reuse of HPC and CI training resources through a *federated training catalog* architecture. The catalog allows providers to retain ownership while publishing standardized metadata to a shared repository [8]. Using an established search backend (Globus Search) and a minimal metadata set aligned with FAIR principles [5, 7], HPC-ED enables both centralized and embedded discovery interfaces [10]. Since its inception, HPC-ED has completed a **pilot phase** demonstrating the feasibility of the federated approach, including:

- **Metadata framework:** Adopting and extending community metadata standards for training materials, including the Research Data Alliance's recommended minimal set and Dublin Core Learning Resource Type [3, 7].
- **Architecture prototype:** Share and discovery workflow using Globus Search with JSON-formatted records [6].
- **Community** – Conducting workshops, tutorials, and hackathons to gather feedback and grow early adopters [8].
- **Planned scaling improvements** – Designing a decentralized "pull-model" architecture for production deployment, integrating quality checks and diverse client tools [10].

https://doi.org/10.22369/issn.2153-4136/17/1/5

1.2 Quality Assurance Automation

While metadata publication processes in the HPC-ED project have matured, **quality assurance (QA)** remains largely manual, relying on human review and link checking. Given the scale of the catalog and the dynamic nature of online resources, automating QA is critical for sustaining trust, maintaining relevance, and enabling scalable federation.

In this paper, we present a proof of concept project that investigates the use of commercial LLMs to automate QA recommendations on metadata quality. Our approach has been to start with a curated subset of the catalog and perform a 2 stage AI assisted review of submitted metadata. The first crawls the URL provided in the metadata to summarize the content, while the second uses an agent to review the metadata in comparison to the summary.

2 METHOD

2.1 Overview

The HPC-ED Beta catalog, as of March 1, 2025, was used as the test set. It contained 128 entries, 5 of which included a YouTube video, either linked or embedded, within a crawl depth of two.

Each item was parsed using BeautifulSoup with Python's built-in `html.parser` [9]. A custom crawler scraped text and URLs from each entry's root metadata URL, descending to a maximum depth of two. For YouTube links, the `youtube_transcript_api` library extracted transcripts [2].

All crawled text was summarized using GPT-3.5 Turbo. A separate review agent, implemented via OpenAI API calls, assessed metadata alignment and provided scoring with rationale. This agent was evaluated across four models: GPT-3.5 Turbo, GPT-4o Mini, GPT-4.1 Nano, and GPT-4.1 Mini.

For each item in the test set, all crawled text was passed to a summarization agent using OpenAI GPT 3.5 turbo. Summaries were stored for each item in the test set for further analysis.

A separate review agent was developed to compare the submitted metadata for each entry against both the catalog description and the crawled summary. The agent produced a score assessing the alignment between the submitted metadata and the content, along with a brief rationale. This review agent was implemented using OpenAI API calls and tested with four models: GPT-3.5 Turbo, GPT-4o Mini, GPT-4.1 Nano, and GPT-4.1 Mini.

2.2 Metadata Review Agent

To evaluate metadata quality, we developed an automated review pipeline implemented in `Python`. After crawling each catalog entry (depth ≤ 2) and summarizing its content using `gpt-3.5-turbo`, we passed the original metadata and summary to a validation agent. The agent was prompted to rate the accuracy of each metadata field, provide reasoning, and, if necessary, suggest improvements. We enforced structured responses in strict JSON format for downstream parsing. Below is the exact prompt used:

```
prompt = f"""
Evaluate the accuracy of the keywords metadata tag on a scale of 1-5
(1 = incorrect, 5 = fully accurate).
If the score is 2 or lower, suggest a better list of keywords based on
the abstract and summary.

Metadata:
Title: {metadata.get('Title', '')}
URL: {metadata.get('URL', '')}
Keywords: {metadata.get('Keywords', '')}
Abstract: {metadata.get('Abstract', '')}
Summary: {metadata.get('Summary', '')}
Duration: {metadata.get('Duration', '')}
Learning Resource Type: {metadata.get('Learning Resource Type', '')}
Expertise Level: {metadata.get('Expertise Level', '')}

Respond in this JSON format:
{
    "Keywords": {
        "score": X,
        "reason": "reason here",
        "suggested_keywords": ["keyword1", "keyword2", "keyword3"]
    }
}
"""
```

A representative output (simplified for brevity) is shown below. (Note that the AI agent treated the additional GPT 3.5 summary as a catalog metadata item to review instead of as ancillary material to guide the review – this will be modified in future versions.)

```
{
  "Title": {
    "score": 4,
    "reason": "The Title field is filled and descriptive, but lacks
    specific details."
  },
  "URL": {
    "score": 3,
    "reason": "The URL is included but should ideally point to the
    specific resource."
  },
  "Keywords": {
    "score": 2,
    "reason": "Keywords are missing, which are essential for
    searchability and categorization."
  },
  "Abstract": {
    "score": 4,
    "reason": "The Abstract provides a concise summary but
    could be more detailed."
  },
  "Summary": {
    "score": 3,
    "reason": "Summary is similar to an abstract; score
    could be higher if it's distinct and adds value."
  },
  "Duration": {
    "score": 1,
    "reason": "Duration is missing, which is crucial for
    learners to estimate time commitment."
  },
  "Learning Resource Type": {
    "score": 5,
    "reason": "Clearly specifies the type of learning resource,
    well-defined."
  },
  "Expertise Level": {
    "score": 4,
    "reason": "Provides guidance on the expected expertise
    level of the audience, but could be more detailed."
  }
}
```

This structured output was parsed and appended to the dataset, allowing quantitative analysis of metadata accuracy across all entries.

3 RESULTS

3.1 Summarization

An example summary of a crawled item, *Python for High Performance* (`https://cvw.cac.cornell.edu/python-performance`), is shown below. The original keywords were: Python, performance, compiling, software, extension modules, algorithms, parallelization, optimization.

Catalog Description Python is a very popular programming language for scientific computing, due to both the expressiveness of the language itself and the availability of a rich ecosystem of packages, tools, and libraries that have been developed by the community to support a wide array of different computational tasks. Python is an interpreted language, however, and therefore Python programs are intrinsically slower than equivalent programs written in a compiled language. This roadmap introduces packages, tools, and strategies that are useful for achieving high computational performance with Python, both on workstations and on multiprocessor clusters.

AI Summary The content discusses Python's popularity in scientific computing due to its expressiveness and ecosystem of packages. It emphasizes achieving high computational performance with Python on workstations and clusters. The workshop assumes prior experience in Python, UNIX/Linux, and general programming concepts. It targets scientists and engineers interested in improving computational performance. Readers can refer to additional resources for Python introduction. System requirements vary from laptops to High Performance Computing systems, and access to relevant packages is necessary for running code examples.

3.2 Keyword evaluation

As part of the automated QA process, keyword quality was evaluated for each crawled item by multiple models. Models were prompted to score the accuracy of the original keywords, provide a short rationale, and suggest an improved set of keywords.

For item *Python for High Performance*, the keyword evaluation scores ranged from 3 to 4 across the four tested models. GPT-3.5 and GPT-4.1 nano each scored the original keywords a 3, noting that while the terms were generally relevant, they lacked specificity for high-performance or scientific computing. Suggested improvements from these models included adding terms such as "scientific computing," "optimization," "compilation," and "clusters." GPT-4.1 mini and GPT-4o both assigned a score of 4, finding the keywords largely appropriate but suggesting more precise language and better coverage of HPC concepts.

3.3 Comparison of sample items with similar model results, keyword evaluation

In the test set, 23 items had the same score value returned by the 4 different models tested. 5 here are sampled for review, and the submitted description and keywords are shown, as well as the AI generated summary of the crawled page. Each models score and reasoning is shown.

Item Title: Data Transfer

Catalog Description: Transferring data and code between your workstation and a remote computer is a common part of scientific workflows. Sometimes this data can be quite large, and sometimes you wish to transmit your data securely. And recently, data transfers between cloud storage and computing facilities are becoming increasingly common. There are a number of utilities available to help you accomplish these essential tasks. Your choice of data transfer utility will depend on how much data you are transferring, how you prefer to perform the transfer, and your priorities (including transfer speed, ease of use, security and validation). This topic presents several data transfer options and their pros and cons, as well as ways to make these transfers faster. While the file transfer techniques presented here are useful in many situations, the included examples will use TACC's Stampede2 and Frontera as the remote computers.

Submitted Keywords: data transfer, security, SCP, performance, strategies, data syncing, data compression

AI Summary: The content discusses the importance of transferring data and code between your workstation and a remote computer, especially in scientific workflows. It mentions the significance of securely transmitting large data and the increasing trend of data transfers between cloud storage and computing facilities. Different utilities are available for data transfers, and the choice depends on factors like the amount of data, preferred transfer method, and priorities such as speed, security, and validation. The topic covers various data transfer options, their pros and cons, and methods to make transfers faster. Specific examples include using TACC's Stampede2 and Frontera as remote computers. The workshop aims to improve participants' ability to transfer files efficiently and choose appropriate transfer utilities based on the situation.

AI Model: 3.5
AI Score: 4
AI Reason: The keywords data transfer, security, SCP, performance, strategies, data syncing, data compression are fairly accurate based on the abstract and summary.
AI Model: 4.1 nano
AI Score: 4
AI Reason: The keywords accurately cover key aspects such as data transfer, security, SCP, performance, data syncing, and data compression, which are relevant to the abstract and summary. However, they could include more specific terms like 'cloud storage' and 'utilities' to better reflect the content.
AI Model: 4.1 mini
AI Score: 4
AI Reason: The current keywords mostly reflect the subject matter of data transfer, security, and performance aspects discussed in the abstract and summary. However, 'SCP' and 'data compression' are not explicitly mentioned or emphasized in the summary or abstract, while other important terms like 'cloud storage', 'file transfer utilities', 'validation', and 'scientific workflows' are missing.
AI Model: 4o mini
AI Score: 4
AI Reason: The keywords largely represent the main topics discussed in the abstract and summary, including key considerations for data transfer. However, they could be more specific to the context of scientific workflows and cloud computing.

Item Title: AI with Deep Learning

Catalog Description: Deep learning comprises a set of methods for Machine Learning and Artificial Intelligence, based on the use of multilayer neural networks to carry out learning. Deep learning techniques can identify patterns in data even within large data sets, and often require substantial computational resources for training model parameters and making predictions. The Frontera supercomputer at the Texas Advanced Computing Center (TACC) is built to support large computational workloads such as those involved with deep learning. Software packages such as TensorFlow, Keras, and PyTorch are widely used to build deep learning pipelines.

Submitted Keywords: deep learning, machine learning, artificial intelligence, TensorFlow, TACC, Frontera, PyTorch, parallelism

AI Summary: Deep learning is a method within Machine Learning and Artificial Intelligence that uses multilayer neural networks to enable learning and pattern identification in data, often requiring substantial computational resources. Frontera supercomputer at TACC supports deep learning workloads, with software packages like TensorFlow, Keras, and PyTorch commonly used. Prospective users of Frontera are typically familiar with HPC and running jobs on clusters. The workshop does not have formal prerequisites, but some familiarity with Linux is recommended. Running example codes requires access to systems with TensorFlow and PyTorch or the ability to install them in a Python virtual environment. Access to supercomputers at TACC, such as Frontera, requires an allocation to the system.

AI Model: 3.5
AI Score: 4
AI Reason: The keywords metadata tag is mostly accurate and relevant to the content of the abstract and summary.
AI Model: 4.1 nano

AI Score: 4
AI Reason: The keywords accurately cover major topics such as deep learning, machine learning, AI, and software tools like TensorFlow and PyTorch. 'Frontera' and 'TACC' are relevant as they relate to the infrastructure discussed. However, 'parallelism' is somewhat broad and less emphasized specifically, and 'Frontera' and 'TACC' are more about the hardware platform than the core concepts.
AI Model: 4.1 mini
AI Score: 4
AI Reason: The keywords mostly align with the abstract and summary, covering deep learning, machine learning, artificial intelligence, TensorFlow, TACC, Frontera, PyTorch, and parallelism. However, 'Keras' is mentioned in both abstract and summary as a key software package but is missing from the keywords. Also, 'HPC' or 'High Performance Computing' could be relevant given the emphasis on supercomputing resources like Frontera.
AI Model: 4o mini
AI Score: 4
AI Reason: The keywords accurately reflect the core topics of deep learning, machine learning, and the relevant software packages, but could be more specific to the context of the workshop.

Item Title: MPI Point-to-Point

Catalog Description: Point-to-point communication encompasses all the methods MPI offers to transmit a message between a pair of processes. MPI features a broad range of point-to-point communication calls; they differ in subtle ways which can affect the performance of your MPI program. This roadmap details and differentiates the various types of point-to-point communication available in MPI-3.0 and discusses when and how to use each method. We will examine blocking as well as nonblocking communication calls and go through some examples using these methods. All exercises and examples are verified to work on Stampede2 and Frontera. MPI also provides for transmission of messages among groups of processes, which is called collective communication. Collective communication is the subject of a different roadmap. This is the second of five related roadmaps in the Cornell Virtual Workshop that cover MPI. To see the other roadmaps available, please visit the complete roadmaps list.

Submitted Keywords: MPI, message passing, buffer, non-blocking, parallelism, probe, deadlock, persistent communication

AI Summary: The content discusses point-to-point communication in MPI, which involves transmitting messages between pairs of processes. It explains the various methods available in MPI-3.0 for point-to-point communication, including differences that can impact performance. The roadmap provides details on both blocking and nonblocking communication calls, with examples. It also mentions that collective communication, which involves message transmission among groups of processes, is covered in a separate roadmap. Various exercises and examples discussed in the content are verified to work on Stampede2 and Frontera. Completing this roadmap will enable users to use MPI on these platforms effectively.
AI Model: 3.5
AI Score: 4
AI Reason: The keywords metadata tag is mostly accurate and relevant to the content of the abstract and summary.
AI Model: 4.1 nano
AI Score: 4
AI Reason: The keywords accurately reflect key concepts discussed in the abstract, such as MPI, message passing, non-blocking communication, and deadlock. However, they could be slightly refined for completeness and clarity.
AI Model: 4.1 mini
AI Score: 4
AI Reason: The keywords are generally relevant and cover many core concepts related to point-to-point MPI communication such as MPI, message passing, non-blocking, probe, and persistent communication. However, the keyword 'buffer' is somewhat generic and not emphasized in the abstract or summary. Also, 'deadlock' is not explicitly mentioned or discussed as a focus in the description.
AI Model: 4o mini
AI Score: 4
AI Reason: The keywords accurately reflect the main topics discussed in the abstract and summary, but could be more specific in relation to the types of communication methods and performance considerations.

Item Title: MATLAB Programming

Catalog Description: MATLAB is a programming and numeric computing platform used to analyze data, develop algorithms and create models. This roadmap includes topics that provide a basic introduction to MATLAB, discuss how to write MATLAB code that can be compiled for speed, and offer tips on getting the best performance out of MATLAB. The example code in these topics was run and verified using MATLAB 2017a through 2021a. Most of it (outside of Mex and other FFI examples) should also work with slight modifications in GNU Octave.

Submitted Keywords: MATLAB, programming, scripting, compiling, functions, GPU, data types, arrays

AI Summary: MATLAB is a platform for programming and numeric computing used for data analysis, algorithm development, and modeling. This content roadmap covers basic MATLAB introduction, optimizing code for speed, and maximizing performance. The example code is verified on MATLAB versions 2017a to 2021a and can work with minor adjustments in GNU Octave. A MATLAB license is necessary to run the software, and trial licenses are accessible. MATLAB is frequently licensed on high-performance computing systems like those at the Texas Advanced Computing Center (TACC).
AI Model: 3.5
AI Score: 4
AI Reason: The provided keywords are relevant to the content in the abstract and summary. They accurately reflect the topics covered in the MATLAB programming resource.
AI Model: 4.1 nano
AI Score: 4
AI Reason: The keywords accurately reflect core topics discussed in the abstract and summary, such as MATLAB, programming, algorithms, code optimization, performance, and compatibility with GNU Octave. However, some keywords like 'compiling' are less

emphasized, and 'GPU' and 'data types' are only briefly mentioned and not central to the content.
AI Model: 4.1 mini
AI Score: 4
AI Reason: Most keywords accurately reflect the content; however, 'scripting' and 'data types' are less emphasized in the abstract and summary. 'Compiling' is relevant, and 'GPU' is somewhat tangential as GPU usage is not explicitly covered in the abstract. Keywords could be more aligned with data analysis, algorithm development, modeling, and performance optimization aspects highlighted in the description.
AI Model: 4o mini
AI Score: 4
AI Reason: The keywords are generally relevant to the topic of MATLAB programming and cover important aspects such as coding, data handling, and performance optimization. However, they could be more focused on specific use cases and the content roadmap.

Item Title: COMPLECS: Parallel computing concepts

Catalog Description: All users of advanced cyberinfrastructure, whether they develop their own software or use 3rd party applications, should understand fundamental parallel computing concepts. In this webinar we cover supercomputer architectures, the differences between threads and processes, implementations of parallelism (e.g., OpenMP and MPI), strong and weak scaling, limitations on scalability (Amdahl's and Gustafson's Laws) and benchmarking. We also discuss how to choose the appropriate number of compute cores or nodes when running your applications and, when appropriate, the best balance between threads and processes. This webinar does not assume any programming experience and is suited for a wide audience, including current and prospective users of parallel computers, anyone who expects to write a proposal for computer time or those who are simply curious about parallel computing.

Submitted Keywords: Expanse, HPC Training, Industry, TSCC, Voyager

AI Summary: SDSC enables discovery through innovative supercomputers, research computing services, and infrastructure support. They provide AI-enabled data science, cyberinfrastructure solutions, consulting, and training for academic researchers and industry partners. The organization translates research into impacts at national, state, and UC system levels, meeting stakeholder needs and advancing scientific discovery. SDSC bridges gaps through education and training, facilitating the impactful use of data and computational resources.
AI Model: 3.5
AI Score: 2
AI Reason: Some of the keywords provided are relevant such as 'HPC Training', but there are better keywords that can be derived from the abstract and summary.
AI Model: 4.1 nano
AI Score: 2
AI Reason: The current keywords (Expanse, HPC Training, Industry, TSCC, Voyager) do not accurately reflect the core content of the webinar, which focuses on fundamental parallel computing concepts, architectures, and scalability. They are either too broad or unrelated to the specific subject matter.
AI Model: 4.1 mini
AI Score: 2
AI Reason: The current keywords focus on specific HPC systems (Expanse, TSCC, Voyager) and a general term 'HPC Training', which are only tangentially related to the webinar content. They do not accurately capture the core themes of the webinar such as parallel computing concepts, parallelism, supercomputing architectures, and related technologies.
AI Model: 4o mini
AI Score: 2
AI Reason: The current keywords do not accurately reflect the focus on fundamental parallel computing concepts, supercomputer architectures, and training for users of advanced cyberinfrastructure.

3.4 Comparison of sample items with different model results, keyword evaluation

The 5 catalog items with the greatest standard deviation between the 4 model scores are listed below.

Item Title: Parallel I/O

Catalog Description: This roadmap presents basic concepts and techniques that will allow your application to take advantage of parallel I/O to increase throughput and improve scalability. The parallel I/O software stack is introduced from the hardware level on up. Emphasis is placed on the Lustre parallel file system, and on MPI-IO as a fundamental API for enabling parallel I/O. These are the building blocks of typical HPC software stacks, including those available on the HPC systems at TACC.

Submitted Keywords: parallel I/O, message passing, file systems, storage, Lustre, MPI-IO, collective operations, parallelization

AI Summary: The content discusses a roadmap for understanding parallel I/O techniques to enhance application performance by leveraging parallel file systems and MPI-IO. It covers fundamental concepts related to Lustre parallel file system and MPI-IO, commonly used in HPC environments like TACC systems. Prior knowledge of Linux shell commands, parallel programming, and MPI is assumed, with suggested roadmaps for prerequisites. Programming experience in C or Fortran is recommended, with foundational roadmaps available. Stampede3 or Frontera systems are used for examples and exercises, requiring specific system access. The roadmap is a continuation of the MPI Advanced Topics content but is not a mandatory prerequisite.
AI Model: 3.5
AI Score: 3
AI Reason: The keywords provided are relevant to the content mentioned in the abstract and summary, but they could be more specific and targeted.
AI Model: 4.1 nano
AI Score: 4
AI Reason: The keywords largely align with the abstract's focus on parallel I/O, file systems, MPI-IO, and Lustre in HPC contexts. Minor elements like message passing and collective operations are

relevant but less emphasized, which explains the slightly lower score.
AI Model: 4.1 mini
AI Score: 5
AI Reason: The keywords accurately reflect the main topics and concepts discussed in the abstract and summary, including parallel I/O, message passing, file systems, storage, Lustre, MPI-IO, collective operations, and parallelization, which are relevant to the content of the resource.
AI Model: 4o mini
AI Score: 5
AI Reason: The keywords relevantly capture the main topics discussed in the abstract and summary, including important terms like Lustre and MPI-IO, which are central to the content.

Item Title: HPC Workshop: MPI

Catalog Description: This two day workshop is intended to give C and Fortran programmers a hands-on introduction to MPI programming. Both days are compact, to accommodate multiple time zones, but packed with useful information and lab exercises. Attendees will leave with a working knowledge of how to write scalable codes using MPI – the standard programming tool of scalable parallel computing.

Submitted Keywords: MPI

AI Summary: A two-day workshop aims to provide C and Fortran programmers with a practical introduction to MPI programming, focusing on writing scalable codes for parallel computing. The workshop features hands-on activities and attendees will gain knowledge on using MPI effectively. The slides provided are from the latest MPI workshop, and inquiries about the workshop can be directed to *Name deleted*.
AI Model: 3.5
AI Score: 4
AI Reason: The keyword 'MPI' is accurate and relevant based on the abstract and summary.
AI Model: 4.1 nano
AI Score: 3
AI Reason: The keyword 'MPI' is relevant but too broad; the metadata lacks specific keywords related to parallel programming, HPC, C, Fortran, or workshop training.
AI Model: 4.1 mini
AI Score: 2
AI Reason: The current keyword 'MPI' is too broad and generic given the detailed context of the workshop. It does not fully capture the scope, audience, or programming languages involved in the workshop.
AI Model: 4o mini
AI Score: 2
AI Reason: The keyword 'MPI' is too narrow and does not capture the comprehensive content of the workshop or the programming languages involved.

Item Title: Data Analysis with R

Catalog Description: This lecture presents several examples of data analysis and visualization in R. It will demonstrate a variety of analyses intended to help researchers determine if learning R is a good investment for their research, including new data analysis examples drawn from the WCM community.

Submitted Keywords: data-analysis, visualization, R, RStudio, data-science, statistics, training, programming

AI Summary: *Name deleted*, a computational scientist with the Cornell University Center for Advanced Computing, discussed data analysis in R, focusing on the Tidyverse and ggplot. He shared examples of using R for data analysis, introduced the importance of data types, variables, functions, and data frames in R. He also talked about how to import and manipulate data, generate summary tables, and create plots using ggplot, including examples of cumulative total calculations and choropleth maps. Additionally, he mentioned the benefits of using literate analysis tools like Quarto Notebooks for reproducibility and interactive plots like Plotly for exploration. *Name deleted* emphasized the significance of factors like the pipe operator for efficient code writing and recommended resources for further learning in R analysis.
AI Model: 3.5
AI Score: 3
AI Reason: The keywords data-analysis, visualization, R, RStudio, data-science, statistics are relevant based on the abstract and summary provided.
AI Model: 4.1 nano
AI Score: 3
AI Reason: The keywords include general terms related to R and data visualization but lack specific references to the key tools and concepts discussed, such as Tidyverse, ggplot, data frames, and Reproducibility tools like Quarto and Plotly, which are central to the abstract and summary.
AI Model: 4.1 mini
AI Score: 5
AI Reason: The keywords accurately reflect the content of the lecture, covering data analysis, visualization, R programming, RStudio, data science, statistics, training, and programming, which align well with the abstract and summary.
AI Model: 4o mini
AI Score: 4
AI Reason: The keywords are largely accurate and relevant to the content of the video, though there could be more focus on specific tools and concepts mentioned in the summary.

Item Title: HPC Workshop: Shared Memory Programming Using OpenMP

Catalog Description: This workshop is intended to give C and Fortran programmers a hands-on introduction to OpenMP programming. Attendees will leave with a working knowledge of how to write scalable codes using OpenMP. Hands-on exercises are included to give attendees practice with the concepts presented.

Submitted Keywords: OpenMP

AI Summary: The workshop provides C and Fortran programmers with a practical introduction to OpenMP programming, focusing on writing scalable codes using OpenMP. Attendees participate in hands-on exercises to reinforce the concepts presented. Contact *Name deleted* at *Name deleted* for any inquiries.
AI Model: 3.5
AI Score: 5
AI Reason: The keyword 'OpenMP' accurately reflects the main topic of the workshop.
AI Model: 4.1 nano
AI Score: 3
AI Reason: The keyword 'OpenMP' is relevant but too broad; the metadata would benefit from more specific keywords such as 'Shared Memory Programming,' 'Parallel Computing,' 'C Programming,' and 'Fortran.'
AI Model: 4.1 mini
AI Score: 3
AI Reason: The keyword 'OpenMP' is relevant but too general and singular given the content focuses on shared memory programming, C and Fortran languages, and hands-on training. More specific keywords could provide better metadata accuracy.
AI Model: 4o mini
AI Score: 2
AI Reason: The current keyword 'OpenMP' is too narrow and does not capture the full scope of the workshop's content which includes programming languages and concepts related to scalable coding.

Item Title: MPI One-Sided Communication

Catalog Description: One-sided communication methods were added to MPI as a part of the MPI-2 improvements and were greatly expanded in MPI-3 by including support for shared memory windows, windows with dynamically attached memory, request-based communication calls, and more window locking mechanisms. On Stampede2 and Frontera, the one-sided communication methods implemented in the Intel MPI and MVAPICH2 libraries use the Remote Direct Memory Access (RMA or RDMA) functionality provided by low-latency interconnect fabrics such as Omni-Path and InfiniBand. In this roadmap, we will introduce the various components of MPI RMA and how to use them. All exercises and examples are verified to work on Stampede2 and Frontera. This is the fifth of five related roadmaps in the Cornell Virtual Workshop that cover MPI. To see the other roadmaps available, please visit the complete roadmaps list.

Submitted Keywords: MPI, communication, window, synchronization, remote memory access, collective operations, fence, buffer

AI Summary: MPI has added one-sided communication methods as part of its improvements, with expanded support for shared memory windows, dynamically attached memory, request-based communication calls, and various window locking mechanisms in MPI-3. These methods are implemented in Intel MPI and MVAPICH2 libraries on systems like Stampede2 and Frontera using Remote Direct Memory Access (RMA) functionality. This roadmap introduces various components of MPI RMA and provides exercises and examples verified to work on Stampede2 and Frontera. It is the fifth out of five related roadmaps in the Cornell Virtual Workshop covering MPI. The completion of this roadmap should enable users to understand and utilize MPI RMA features effectively, with specified system requirements.
AI Model: 3.5
AI Score: 5
AI Reason: The keywords provided align well with the content of the abstract and summary.
AI Model: 4.1 nano
AI Score: 3
AI Reason: The keywords broadly cover MPI, communication, and synchronization but lack specificity about the focus on Remote Memory Access (RMA) and shared memory windows, which are central topics in the abstract and summary.
AI Model: 4.1 mini
AI Score: 4
AI Reason: The keywords generally cover the main topics such as MPI, communication, window, synchronization, remote memory access, fence, and buffer. However, 'collective operations' is less relevant here since the abstract and summary focus primarily on one-sided communication and Remote Memory Access (RMA) mechanisms rather than collective operations. Also, adding 'RMA', 'one-sided communication', 'Intel MPI', and 'MVAPICH2' would improve relevance.
AI Model: 4o mini
AI Score: 2
AI Reason: The current keywords do not fully encompass the specific aspects of one-sided communication methods introduced in MPI-2 and MPI-3, such as request-based communication, shared memory windows, and RMA functionality, which are significant in the abstract and summary.

4 DISCUSSION

As a summarization agent, GPT 3.5 was sufficient to create reasonable summaries of crawled content and submitted metadata, overlooking HTML formatting and general page boilerplate, to access meaningful content on the page. Both catalog metadata and crawled content were fed into the summarization agent, it might be interesting in future to look at whether catalog metadata as an input improves summarization ability relative to crawling alone.

Additionally, a substantial number (23 of 128 items) had the same scores between the models. This combined with the ability of smaller models to distinguish between content and non-content when summarizing crawled media does indicate that there is strong potential for the use of commercial unmodified LLMs for automation of metadata review.

While overall, the quality of agent responses scales as expected with the size and capability of the LLM, in some cases we saw higher level model "overthinking" analysis. Consider HPC Workshop: Shared Memory Programming Using OpenMP under the differing results. The catalog item is for a workshop on OpenMP. The submitted keyword is "OpenMP." GPT-3.5 found this to be an adequate keyword, whereas GPT-4.1 and GPT-4o scored it lower with the reason that the keyword was too vague, wanting the languages used in the workshop and additional training also included.

We noticed that stronger models tended to favor more expansive keyword sets, which could harm discovery by failing to focus on the core principles of each catalog item. The larger models created suggested keyword lists with more depth and penalized submitted metadata for the lack of it, but in many cases simpler keywords were a feature rather than a flaw. This poses a challenge for further automation of metadata review, one which we hope can be mitigated through better prompt engineering. It may also be helped through other efforts to define an ontology for HPC training.

Additionally, we would like to include direct comparisons with LLM based metadata review and human metadata review in the future.

5 RELATED WORK

Members of the HPC-ED team have been using HPC-ED metadata to develop LLMs as part of their training curricula. These include the SDSC HPC/CI Training Catalog LLM that uses the Microsoft GraphRAG model [4]; the Texas Advanced Computing Center built a Retrieval-Augmented Generation (RAG) application that scrapes the HPC-ED federated catalog, using Sentence Transformers for semantic search, then generating a personalized, step-by-step curricula from natural language queries. The system ranks relevant modules, suggests prerequisites and follow-up topics, and supports HTML/PDF export, helping students and instructors quickly connect questions to authoritative HPC learning materials while lowering barriers for beginners. The Cornell University Center for Advanced Computing is also developing a RAG agent but which is based on Cornell Virtual Workshop (CVW) training materials; starting with an AI-enabled search function, which will accept natural language queries, including the level and depth of material sought in addition to topic or goal, and output a simple answer as well as link(s) to CVW learning resources.

The LLMs will be tested using the crawler described above.

ACKNOWLEDGMENTS

This work builds on the efforts of all of the partners of the HPC-ED project and is supported by CyberTraining: Pilot: HPC ED: Building a Federated Repository and Increasing Access through CyberTraining – NSF grant OAC-2320977.

REFERENCES

[1] K Cahill, D Joiner, S Lathrop, S Mehringer, and A & Navarro, J-P & Weeden. 2022. Final Results: National Survey on Educational and Training Materials Repositories. https://www.cac.cornell.edu/about/pubs/Survey2022.pdf

[2] Johannes Depoix. 2025. YouTube Transcript API. https://github.com/jdepoix/youtube-transcript-api. Accessed: 2025-08-11.

[3] Dublin Core. 2043. LRMI Learning Resource Type vocabulary. Retrieved September 8, 2024 from https://www.dublincore.org/specifications/lrmi/concept_schemes/learningResourceType

[4] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2025. From Local to Global: A Graph RAG Approach to Query-Focused Summarization. arXiv:cs.CL/2404.16130 https://arxiv.org/abs/2404.16130

[5] Leyla Garcia, Bérénice Batut, Melissa L. Burke, Mateusz Kuzak, Fotis Psomopoulos, Ricardo Arcila, Teresa K. Attwood, Niall Beard, Denise Carvalho-Silva, Alexandros C. Dimopoulos, Victoria Dominguez Del Angel, Michel Dumontier, Kim T. Gurwitz, Roland Krause, Peter McQuilton, Loredana Le Pera, Sarah L. Morgan, Päivi Rauste, Allegra Via, Pascal Kahlem, Gabriella Rustici, Celia W.G. Van Gelder, and Patricia M. Palagi. 2020. Ten simple rules for making training materials FAIR. *PLoS Computational Biology* 16, 5 (2020), 1–9. https://doi.org/10.1371/journal.pcbi.1007854

[6] Globus. 2024. Globus Search Service. https://docs.globus.org/api/search/.

[7] N. J. Hoebelheinrich, K. Biernacka, M. Brazas, L. J. Castro, N. Fiore, M. Hellström, E. Lazzeri, E. Leenarts, P. M. Martinez Lavanchy, E. Newbold, A. Nurnberger, E. Plomp, L. Vaira, and A. van Gelder, C. W. G. & Whyte. 2022. Recommendations for a minimal metadata set to aid harmonised discovery of learning resources. https://doi.org/10.15497/RDA00073

[8] Susan Mehringer, Mary P Thomas, Charlie Dey, Kate Cahill, David Joiner, Richard Knepper, and Jeaime H Powell. 2023. Scaling HPC Education. In *Tenth SC Workshop on Best Practices for HPC Training and Education BPHTE23*. Denver, 41–46. https://doi.org/10.22369/issn.2153-4136/15/1/8

[9] Leonard Richardson. 2024. Beautiful Soup Documentation. https://www.crummy.com/software/BeautifulSoup/bs4/doc/. Accessed: 2025-08-11.

[10] Mary P Thomas, Susan Mehringer, Katharine Cahill, Charlie Dey, Brian Guilfoos, David Joiner, John-Paul Navarro, Jeaime H Powell, and Richard Knepper. 2024. Building a Federated Catalog for CyberTraining Materials: The HPC-ED Pilot Project. In *Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* (New York, NY, USA), Vol. 1. ACM, 1–5. Issue 1. https://doi.org/10.1145/3626203.3670586

A Retrospective on South Africa's Student Cluster Competition and its Model for Inclusive HPC Outreach and Training (2012-2020)

Bryan Johnston
CSIR
bjohnston@csir.co.za

Nick Thorne
Austin TX
nick.thorne@dell.com

Matthew Cawood
TACC
mcawood@tacc.utexas.edu

Eugene de Beste
Cape Town, South Africa
eugene@debeste.co.za

David Macleod
CSIR
dmacleod@csir.co.za

John Poole
Clemson University
jopoole@clemson.edu

ABSTRACT

The Centre for High Performance Computing (CHPC) is South Africa's national supercomputing facility. In 2012, it launched an outreach initiative to raise awareness of High-Performance Computing (HPC) among undergraduate students through the creation of the Student Cluster Competition (SCC). A national contest was designed to train and showcase student talent in a spirited, hands-on environment. The initial stage of the CHPC SCC saw twenty teams of four undergraduate students undergo an intensive week of HPC training, covering Linux fundamentals, cluster design, and system administration. Finalists from this selection round would then compete in a live challenge using HPC systems of their own design, with the top competitors selected to represent the CHPC at the International Student Cluster Competition hosted at the ISC High Performance conference in Germany.

From its inception, the CHPC SCC has prioritised demographic diversity and equal opportunity, actively recruiting students from historically disadvantaged communities to ensure inclusive participation and representation. A rapid teaching framework was developed to address key knowledge gaps in HPC system design, administration, and optimisation: the empowerment of students with limited prior exposure in the field of HPC to excel. This approach has proven highly effective: South African teams ranked in the top three internationally for eight consecutive years, demonstrating the strength of the program.

This paper presents the strategy and structure behind the CHPC SCC, detailing the training model, selection process, and evaluation methods used for both national and international rounds. It highlights how the initiative has evolved into a recognised platform for HPC education, enabling students to learn about HPC and become global contenders in the field.

https://doi.org/10.22369/issn.2153-4136/17/1/6

KEYWORDS

HPC Education, Student Cluster Competition, Diversity in HPC, Best Practices in Training, International Cooperation, Workforce Development, South Africa

1 INTRODUCTION

The Centre for High Performance Computing (CHPC)[1] is South Africa's national supercomputing facility. It was established to support computational research in scientific and industrial fields. In 2012, the CHPC identified a critical and persistent skills gap: undergraduate students received little to no theoretical or practical exposure to High-Performance Computing (HPC). The gap was evident across the South African tertiary education landscape, but it was most pronounced among students from historically disadvantaged institutions (HDI's) and under-served communities. These groups in particular faced compounded barriers - limited access to computing infrastructure, scarce opportunities for specialist training, and systemic exclusion from emerging computational fields.

To bridge this gap, the CHPC launched the Student Cluster Competition (SCC), a national initiative designed to raise awareness, cultivate core technical competencies, and establish a pipeline of future HPC practitioners within the country. The SCC model introduced a novel blend of hands-on education and competitive participation, positioning HPC as a powerful research instrument and a dynamic, team-based discipline accessible to talented undergraduate students.

By combining merit-based selection with inclusive outreach, the SCC reimagined HPC as a field in which students from diverse backgrounds in South Africa could not only gain exposure to advanced computing technologies, but also earn the opportunity to represent their country on the global stage at an international Student Cluster Competition.

2 HISTORY

In 2011, at the Association for Computing Machinery (ACM) and Institute of Electrical and Electronics Engineers (IEEE) Computer Society's SC conference (formerly Supercomputing)[2], the HPC Advisory Council[3] introduced the CHPC to its newly established Student Cluster Competition (SCC) programme, which was due to

[1]CHPC homepage: https://www.chpc.ac.za
[2]SC'11: https://sc11.supercomputing.org/
[3]HPC Advisory Council: https://www.hpcadvisorycouncil.com/

Table 1: Phased Development of the CHPC Student Cluster Competition Programme (2012–2021)

Phase	Years	Characteristics	Representative Milestones
Establishing	2012–2014	Programme launched without a prior model. Structure built from first principles with improvised materials and initial outreach.	2012: first Winter School and national competition. 2013: ISC SCC Overall Champions at first attempt. 2014: ISC SCC Overall Champions.
Consolidation	2015–2018	General framework stabilised. Growing institutional participation and consistent performance.	2015: ISC SCC 2nd place. 2016: ISC SCC Overall Champions. 2017: ISC SCC 2nd place. 2018: ISC SCC 3rd place. CHPC SCC introduces Intel-sponsored award for best female student.
Maturity and Handover	2019–2021	Operational maturity established. Leadership transition began, sponsor support deepened and innovation slowed.	2019: ISC SCC Overall Champions. Major sponsorships secured. 2020: ISC SCC 2nd place. 2020–21: virtual and hybrid adaptations during handover.

commence in 2012 at the ISC High Performance conference[4], formerly the International Supercomputing Conference. As a new initiative, the HPC Advisory Council actively promoted the competition and recruited teams from across the world. Given that Africa had not previously been represented in the SCC, the Council extended an invitation to the CHPC to participate in the inaugural event.

Owing to strategic considerations, the CHPC declined the invitation. Two critical constraints were immediately evident: the near total absence of HPC education and training at South African universities, and the limited time available to prepare a team capable of meaningful participation. Instead, the CHPC proposed spending the next year and a half building a national SCC programme, with a focus on identifying and training students to prepare a team for participation in the 2013 ISC Student Cluster Competition (ISC SCC). This proposal was endorsed by the HPC Advisory Council.

This approach gave the CHPC sufficient time to decide what it wished to achieve from participation in the ISC SCC, rather than making participation itself the sole goal. Table 2 summarises the identified key objectives:

Table 2: CHPC SCC Identified Objectives

- Expose as many students as possible to HPC
- Increase participation by students from previously disadvantaged communities
- Increase participation by women
- Prepare teams to be competitive at the ISC SCC

As a national supercomputing facility, the CHPC primarily serves the South African higher education sector and does not maintain its own student body. It therefore needed to collaborate directly with universities, and this engagement required a transparent approach to ensure equitable access for all institutions.

Within two years the structure produced international success. In all, CHPC teams captured the ISC SCC Overall Championship in 2013, 2014 and 2016. They also achieved podium finishes in 2015, 2017, 2018, 2019, and 2020. This level of achievement, attained with rotating undergraduate teams where few students competed more than once, demonstrated both elite performance and wide distribution of experience across the student community [8].

3 PROGRAMME PHASES

The evolution of the CHPC's SCC over its first decade can be described in three developmental phases, each marked by specific organisational priorities and outcomes. This phased view highlights how the programme matured from inception to operational stability while maintaining a consistent educational mission. Table 1 provides a consolidated overview.

The **first phase** was characterised by experimentation and credibility-building, with a rapid transition from design to international recognition. The **second phase** marked consolidation, with the CHPC becoming a reliable pipeline of competitive teams and embedding inclusivity through its rotation policy. The CHPC mandated undergraduate-only entry from the outset, thereby maximising the number of students able to attend ISC and allowing the national delegation to include students from more than one university; further operational implications are discussed in the National Round section. The **third phase** represented operational maturity, in which the programme had stabilised and secured strong sponsorship, but began to show signs of needing strategic renewal. The leadership handover during this time created the conditions for the next era, which falls outside the scope of this account.

4 PROGRAMME DETAILS

The CHPC established a structured competition programme culminating in international representation at ISC SCC, as summarised in Table 3. Teams progressed through multiple stages, with participation limited to South African universities after initial attempts to

[4]ISC SCC: https://isc-hpc.com/program/student-cluster-competition/

Table 3: Structured Annual Progression of the CHPC Student Cluster Competition

Round	Timeline	Focus	Purpose / Outcome
Selection Round	January to July	Inclusive, low-cost IT, Linux, and HPC exposure; theory and practical training	Raise awareness of HPC; build foundational skills; identify promising student teams.
National Round (CHPC SCC)	July to December	Small scale ISC SCC-like competition; hands-on cluster assembly; benchmarking and testing	Simulate ISC SCC experience; select national team; prepare for international representation.
International Round (ISC SCC)	December to June (next year)	ISC SCC preparation and participation; intensive technical refinement; team coordination and tuning	Represent South Africa at ISC SCC; compete internationally; showcase national HPC talent.

include broader Southern African institutions proved logistically unfeasible. The framework underpinning the programme's delivery encompasses the organising committee, its pedagogical approach, and commitments to diversity and inclusion. Later subsections (Selection Round, National Round, International Round) provide detailed accounts of each competitive stage.

4.1 Organising Committee and Operations

The programme's sustained success stemmed from a collaborative leadership structure centred on the Advanced Computer Engineering (ACE) Lab, which serves as the CHPC's research and development unit. The organising committee combined strategic oversight with operational expertise, integrating technical, pedagogical, and logistical responsibilities under unified leadership. Collective experience and institutional memory proved instrumental in maintaining programme identity and operational stability.

Leadership: The ACE Lab head provided strategic continuity across multiple cycles, supported by engineers and technologists whose expertise spanned systems and cluster architecture, benchmarking, Linux systems, and pedagogy. This core ensured consistent standards while adapting to evolving competition requirements.

Lectures: Instruction was delivered by domain specialists and co-presenters, balancing technical depth with skill development. Guest lecturers from HPC-adjacent fields contributed expertise in areas such as benchmarking methodologies and scientific applications. The programme also created opportunities to upskill staff beyond the ACE Lab, extending institutional capacity.

Logistics: The programme's logistics were demanding, as multiple competition cycles had to be organised in parallel. The Selection Round was co-hosted with the CHPC Winter School[5] and relied on partnerships with host universities to provide lecture venues, computer laboratories, catering, and frequently accommodation. This model significantly reduced both costs and coordination burdens through shared infrastructure. The National Round, held alongside the CHPC's annual conference[6], required teams to assemble physical clusters supplied on loan by hardware sponsors. Equipment was usually delivered only a few days before the competition and returned immediately afterwards, which preserved the authenticity of the exercise while limiting storage and shipping costs. The International Round, hosted by ISC High Performance in Germany, introduced still greater complexity. Hardware specification finalised early in the year was shipped to Cape Town for assembly, exported to Germany for staging in ISC warehouses, and eventually returned to the manufacturer. Despite this intricate sequence, deliveries were consistently completed on time.

Materials development: Curriculum development was closely aligned with the ACE Lab's research mandate. Early initiatives such as the Ranger Project and the HPC Ecosystems Project[7] [3, 5] informed course content and provided training platforms, in addition to their original purpose of equipping the African continent with HPC capability. Internal research on hybrid cloud–HPC management systems and software-defined infrastructure [2, 7] also shaped tutorials and practical exercises for participants. However, automation tools were deliberately excluded to ensure equitable access across institutions with differing resources. Students instead carried out manual deployments, an approach that reinforced fundamental skills while maintaining relevance in diverse institutional environments.

4.2 Pedagogy Framework

The SCC's competitive structure was designed to motivate engagement while supporting educational inclusivity. The challenge lay in ensuring that advanced teams did not outpace novices in ways that limited broader learning. The pedagogical framework addressed this tension through a staged progression: **foundational skills** in the Selection Round, **applied practice** in the National Round, and **synthesis with evaluation** in the International Round. Each stage reinforced the previous, deepening knowledge while preserving accessibility for participants from diverse academic and institutional backgrounds. Detailed curriculum design and delivery are described in the later round-specific subsections.

[5] CHPC Winter School: https://wiki.chpc.ac.za/workshops:practicalhpc

[6] CHPC Annual Conference: https://chpcconf.co.za/

[7] Internationally funded capacity-building initiatives that deployed HPC clusters and training programmes across Southern African universities.

4.3 Diversity and Inclusive Opportunities

Diversity formed an explicit element of the programme's design. Outreach activities included targeted engagement with historically disadvantaged institutions, and selection criteria considered both demonstrated technical performance and indicators of future development potential. Team composition aimed to reflect South Africa's demographic profile while maintaining merit-based progression. In practice this approach resulted in the inclusion of students with limited prior exposure alongside more experienced peers, extending participation across a broader cohort.

At the international stage, team size increased from four to six members in line with competition rules. The CHPC addressed this by supplementing the winning national team with outstanding individuals drawn from runner-up teams, resulting in hybrid delegations that represented multiple institutions. From the CHPC's perspective this structure was a distinctive feature of the programme, enabling broader representation while maintaining competitiveness. Recognition of underrepresented groups was also formalised; for example, in 2018 Intel sponsored an award for the best female student, highlighting the programme's emphasis on equitable participation [9].

4.4 First Round — "Selection Round"

The first round of the CHPC SCC serves as the programme's main entry point and the largest contributor to national impact. It provides students with a shared foundation, supports the identification of promising teams, and begins the pipeline that ultimately leads to international representation at ISC SCC. Participation is restricted to South African universities, and admission is determined through a combination of institution-ranked nominations and CHPC allocation to ensure equitable representation.

Curriculum overview: The week integrates lectures with hands-on laboratory work. Students begin with Linux fundamentals and basic system administration before progressing to clustering concepts that emphasise hardware trade-offs and essential software layers. They then move on to creating virtual clusters and configuring basic networking through a graphical cloud interface, followed by the installation of a standard HPC software stack, including compilers, MPI, and scientific libraries. The programme concludes with performance benchmarking using workloads common to SCC competitions, such as HPL and HPCG. Alongside the technical content, newly formed teams establish working norms and practice collaboration.

Delivery model: A host university provides a lecture venue for morning teaching and a computer laboratory for afternoon practicals. This rhythm supports immediate application of concepts and encourages team-based problem solving. Co-hosting with the CHPC Winter School streamlines logistics and reduces cost through shared facilities and partial sponsorship.

Progression and structure: Learning is scaffolded to align a diverse cohort around a shared baseline, moving from terminology and tooling to system services, parallel programming concepts, and performance measurement. The day-by-day progression and assessment checkpoints are summarised in Table 4. The emphasis is on system-level reasoning rather than mechanical execution, so students document decisions, justify trade-offs, and reflect on failure modes.

Assessment: Teams complete short, graded theory questions that are directly linked to their practical exercises, and progress is monitored against defined milestones to ensure completeness and reproducibility. On Day 1 each team receives a design brief requiring a costed cluster proposal aligned to the theoretical workloads they are assigned. The process culminates on Day 6, when teams present and defend their designs in a timed session before a judging panel that may include past competitors, invited experts, and competition organisers. Evaluation considers benchmark accuracy, the quality of reasoning, and the effectiveness of teamwork. Performance in these areas determines which teams advance to the National Round.

Applications and cohort: Applicants are typically in their second or third year of undergraduate study with strong academic records. First-year students are not considered, as limited prior exposure to computing and a lack of academic grounding make it difficult for them to contribute meaningfully within the intensive timeframe of the competition. Final-year students are also excluded to ensure that selected students remain eligible to represent South Africa the following year at the international round. This policy maximises continuity and strengthens the pipeline by allowing returning participants to advance through successive rounds. Cohorts are intentionally mixed, ranging from experienced programmers to students with minimal prior computing exposure, and the structure, pacing, and mentoring are designed to narrow initial disparities while preserving healthy competition.

4.5 Second Round — "National Round"

The National Round builds directly on the foundations of the Selection Round but places students in a more tightly constrained environment. Held during the CHPC Annual Conference, it requires teams to design, assemble, configure, and validate physical clusters within a compressed timeframe that mirrors the intensity of the ISC SCC format.

Preparation and design: Teams receive a standardised parts list prepared by the CHPC SCC hardware partners, together with a design envelope aligned to the expected applications. Facilitators intervene only in cases of safety concerns or blocking technical issues. Within these boundaries, students make performance-driven choices, manage budget and power constraints, and document the rationale for their decisions.

On-site deployment: Hardware arrives shortly before the event and is returned immediately afterwards. Teams are responsible for the complete deployment process: assembling the systems, installing operating environments, and configuring all required services. In effect, students must act as HPC Linux administrators, diagnosing hardware and software issues under strict time constraints while maintaining a functioning cluster. Beyond bringing the system online, they are also required to compile, install, and tune both synthetic benchmarks and representative scientific workloads. This work extends the skills introduced in the Selection Round and places students in a setting that mirrors the expectations of professional HPC practitioners, where system reliability, performance optimisation, and application readiness must be achieved simultaneously.

Table 4: Learning Progression Across the Selection Round

Day	Main theme	Outcomes / objectives	Assessment method
1	Foundations of HPC and Linux	Recognise core HPC concepts; navigate Linux; basic IPv4 networking	Tutorial lab milestones
2	Hardware and cluster design; debugging	Configure services; understand hardware trade-offs; trace misconfigurations	Lab validation; short written explanations
3	Cluster services; parallel computing	Compile and deploy applications; manage shared services	Tutorial milestones; theory questions
4	Benchmarking and preparation	Execute HPL; finalise theoretical design	HPL execution; design brief
5	Revision and Q&A	Consolidate concepts; reinforce system reasoning	Open Q&A; lab completion
6	Design presentation	Present and defend cluster design; respond to critique	10-minute team talk; 5-minute Q&A

Engagement and communication: In addition to their technical responsibilities, teams interact continuously with judges, conference attendees, and on occasion the media throughout the competition. These interactions take place during the same period as system assembly, configuration, and benchmarking, requiring students to balance communication demands with their technical workload. The format therefore evaluates not only the clarity of explanation, design coherence, and professional conduct, but also the ability to manage time and perform under simultaneous technical and presentational pressures.

Assessment and selection: Required outputs include validated results for synthetic and application-oriented benchmarks, configuration artefacts, and a concise design justification. An external panel evaluates not only the accuracy of results and the quality of reasoning, but also the teamwork and communication demonstrated under competition conditions. Performance across these dimensions determines which students advance to the International Round.

4.6 Final Round — "International Round"

The International Round serves as the capstone and applies the programme's scaffold at full scale. The CHPC formed a six-student undergraduate delegation by supplementing the four-member National Round winning team with two additional undergraduates from other teams so that the delegation represented multiple institutions; in addition, two undergraduate reserves were selected as stand-by members.

Rationale and team composition: From the programme's inception the CHPC required that local teams be undergraduate-only, with four-member teams competing at the National Round. For international participation this winning four was supplemented by two additional undergraduates drawn from other teams, selected as high performers or promising individuals, to form the six-student core delegation permitted by ISC rules. A further two undergraduates were designated as reserves, creating an eight-member national squad in total. Reserves underwent the same pre-departure preparation and skills development, including sessions with engineers at Dell Technologies and staff at the Texas Advanced Computing Center, but only travelled to ISC if a core member was unable to participate. This arrangement maximised opportunity, maintained representation from more than one university, and ensured continuity in the event of unforeseen constraints.

Industry collaboration: Preparation includes design review and skills development with engineers at Dell and staff at the Texas Advanced Computing Center. Sessions cover current HPC technologies, configuration practice, power monitoring and budgeting, and benchmarking methodology. Students present a proposed system design, receive feedback, and iterate before finalisation.

Training and roles: With the design set, students assemble and tune the system, validate workloads, and explore performance limits. Roles emerge naturally, for example application specialist and systems administrator, reflecting how tasks will be divided in competition.

Logistics: The international shipping sequence has several stages. Hardware is finalised early in the calendar year, shipped to Cape Town for assembly and testing, exported to Germany for staging at ISC, and returned to Europe by way of South Africa after the event. The chain demands careful scheduling and coordination but was consistently delivered on time.

Competition: At ISC High Performance the team reassembles the system, verifies stability, and completes assigned workloads within strict time and power limits. Facilitators step back to an advisory role while students execute independently. The round is both a contest and a training loop: alumni return as mentors, feeding experience back into the Selection Round and National Round.

5 ASSESSMENT AND EVALUATION

The effectiveness of the CHPC SCC programme is evaluated using performance standards, participant feedback, and longitudinal review across the following dimensions:

- **Programme throughput and access**: The rotation rule prevents repeat participation at ISC and the international roster is limited to six students per year. The undergraduate-only mandate and four-to-six supplementation were adopted to

adopted to maximise international exposure and broaden institutional representation.
- **Competition performance**: Consistent international results were achieved over 2013–2020, including overall wins in 2013, 2014, 2016 and 2019; second place in 2015, 2017 and 2020; and third in 2018.
- **Participant outcomes**: Alumni have progressed into high-performance computing, data science and engineering, with many taking roles in national research institutes and industry.
- **Diversity impact**: Recent cohorts included more than half historically disadvantaged participants, with representation from all nine provinces, in line with the outreach and selection approach.

Evaluation methods include pre- and post-course questionnaires to track skill development, analysis of competition results against established benchmarks, and longitudinal follow-up of alumni to assess academic and professional trajectories.

6 RELEVANCE AND APPLICATION

The CHPC SCC model has demonstrated value not only in meeting its original objectives but also in broader educational and institutional contexts. Its structured, incremental design provides a flexible framework for HPC training that can be adapted to varied settings and participant backgrounds. Beyond equipping students with technical skills, the model contributes to institutional capacity building, raises awareness of HPC, and fosters collaboration across universities.

Undergraduate participation where HPC exposure is limited: The model lowers barriers to entry and offers a practical pathway into HPC education, particularly in regions where HPC has yet to be embedded in curricula. By enabling students to engage with HPC concepts earlier in their studies, it expands the pipeline of future practitioners and researchers.

Accelerated skill gain across heterogeneous cohorts: The tiered progression accommodates students with widely differing levels of prior knowledge. Even participants who do not progress to the international stage acquire meaningful skills and confidence, ensuring that the benefits extend across the entire cohort.

Motivation through international pathways: The opportunity to compete internationally and attend ISC High Performance has proven to be a powerful motivator. It encourages individual learning, elevates the profile of HPC within institutions, and strengthens institutional support for continued participation.

Collaborative problem-solving: Teams are required to design, configure and benchmark working systems under time and resource constraints. This develops technical competence alongside vital soft skills, including communication, coordination and leadership.

Talent pipeline for research and industry: Graduates of the programme move into computational research, engineering and data-intensive industries with operational experience that shortens the transition from academic training to professional practice. This reinforces the national HPC talent pipeline and supports broader research and innovation goals.

A defining feature of the model is its emphasis on practical, hands-on training. Participants move beyond theory to active engagement in system design, configuration and benchmarking, gaining both conceptual understanding and operational proficiency. These skills are directly transferable to academic research, industry roles and national HPC initiatives, ensuring that the benefits of the programme extend well beyond the competition itself.

7 REFLECTION

7.1 Challenges

Running a competition in a resource-constrained environment presents a unique set of challenges. Students often enter with little prior exposure to high-performance computing, and many have limited awareness that HPC even exists as a career path. Although this lack of awareness was the main motivation for launching the outreach initiative, it also means that each cohort begins with a steep learning curve, with uneven levels of preparation between teams.

Another challenge lies in the long-term sustainability of the pipeline. Although many students benefit greatly from training and competition, only a subset remain connected to the broader HPC ecosystem after completing their studies. Others, despite their initial enthusiasm, disappear from the pipeline, highlighting the difficulty of maintaining momentum without structured follow-up opportunities or consistent mentorship.

The time available for facilitators to interact directly with students is also limited. Throughout the program, facilitators typically have only about a month of face-to-face engagement: approximately one week in Round 1, one week in Round 2, one week in the United States, and one week preparing at CHPC. This does not include the competition itself. The constraint limits the depth of mentoring that is possible and makes it difficult to provide extended support to students who fall behind.

Within competition itself, the disparities in performance are stark. The rankings are often predictable from the outset, with a handful of strong teams steadily improving year after year, while others struggle from the beginning. This gap reflects broader systemic issues: unequal access to local computing resources, uneven levels of supervision and mentorship, and institutional differences in technical capacity. These disparities make it difficult to ensure a level playing field between all participants, despite the competition's efforts to do so.

7.2 Highlights

Despite the difficulties of working in a resource-limited context, the competition has achieved notable successes that demonstrate both its impact and its replicability. First, the programme has received strong institutional and industry support. Sustained funding from the CHPC, combined with international collaborations, has enabled South Africa to field competitive teams year after year. Partnerships with TACC and Dell Technologies have provided technical expertise and cutting-edge competition hardware. The ISC-HPC has also supported the effort by reserving a competition slot for the South African SCC team, ensuring a competition pipeline and consistent exposure on the global stage. These collaborations highlight the importance of external support: emulating this model elsewhere would likewise require investment and commitment from pivotal external stakeholders.

The achievements of participants further illustrate the program's impact. Former students now work in HPC and adjacent fields, both in South Africa and abroad, including several of the authors of this paper. Others have gone on to build lasting communities of practice, such as the University of the Witwatersrand's SIG HPC [1, 6], extending the competition's influence into sustained academic and professional networks. South African teams have earned repeated accolades at ISC, including multiple top-three placements across consecutive years, underscoring the competitiveness of the training model.

Underlying these outcomes is a training curriculum that has been refined over more than a decade of practice. The curriculum integrates best practices such as distributing an introductory data packet with YouTube tutorials prior to the event, which allows students to become familiar with fundamental concepts in advance. During the training sessions, facilitators deliberately design hands-off practicals to encourage peer problem-solving, freeing up time to provide targeted support to teams that are struggling. At the same time, experience has shown that the curriculum's fast pace can leave some students behind, and facilitators are often stretched too thin to fully close these gaps. Nonetheless, the iterative improvements have yielded a repeatable, structured model for rapid HPC training, one that has consistently elevated students from little prior exposure to international competitiveness in just a matter of weeks.

7.3 Lessons Learned

Several broad lessons have emerged from more than a decade of running the CHPC SCC, many of which are applicable to similar initiatives elsewhere. The most consistent finding is that hands-on, competitive formats are uniquely effective. The act of building, breaking, and fixing real systems under time pressure develops confidence and technical maturity much faster than classroom teaching alone. Students remember not the instructions they are given, but the problems they solve.

A second lesson is that student preparation is never uniform. Effective programs must recognize this reality and design scaffolding to raise the floor. Introductory materials, structured mentoring, and targeted one-on-one guidance allow students with less background to contribute meaningfully while still challenging the strongest participants.

Third, access to resources matters. Cloud platforms provide a useful entry point, but they come with limitations in reliability, accessibility, and longevity. Long-term success depends on giving students consistent access to real HPC environments. The 2024 introduction of an OpenHPC virtual lab [4], which coincided with the South African team's return to the top three at ISC, underscored how impactful this access can be.

Finally, sustainability requires more than annual training cycles. Continued funding, active alumni networks, and strong institutional partnerships are essential to keep the pipeline vibrant. Alumni who remain engaged, as mentors, organizers, or professionals in HPC, extend the impact well beyond a single competition.

Perhaps the deepest lesson is philosophical: HPC expertise cannot be handed down fully formed. Mastery comes only through repeated attempts, failures, and refinements. The competition succeeds not because it teaches everything, but because it channels student passion into a structured environment where persistence leads to growth. The best students ultimately learn more on their own than they ever could from direct instruction, and the SCC gives them the spark to begin that journey.

7.4 Future Work

Future plans for the CHPC SCC are shaped by years of hands-on engagement, reflective practice, and the continuous stream of ideas sparked while designing, mentoring, and iterating in real-world HPC training environments.

While the vision continues to evolve, this account serves as a retrospective - capturing the scaffolding, spirit, and lessons of a formative era. Leadership of the SCC initiative has since been passed on, ensuring that new voices and fresh perspectives carry the momentum forward.

8 REPRODUCIBILITY AND RESOURCES

The CHPC SCC was not originally designed as a reproducible turnkey solution for other HPC centres, however, many elements of the model can be adopted elsewhere. Over the years, consistent documentation has been difficult to maintain, particularly as facilitators moved on to other roles, leaving an incomplete archival record. Still, several resources remain accessible: the University of the Witwatersrand's University HPC Special Interest Group (SIG HPC) has documented their student-led training pipeline that prepares teams for SCC and international competitions [1, 6], and video content created for the 2020 competition during the COVID-19 pandemic (available on CHPC's YouTube channel) provides a concrete snapshot of training delivery and expectations that can be adapted by others.

Reproducing the technical infrastructure, modest clusters or cloud resources, standard Linux/MPI stacks, and familiar benchmarks such as HPL and HPCG, is relatively straightforward. The more difficult component to reproduce is the facilitators. At CHPC, a team of full-time engineers took on the role of facilitators during the competition: they planned, coordinated logistics, delivered training, and mentored students, from the fundamentals to competing in Germany. This level of engagement was made possible by official time allocated to the competition, ensuring that professional expertise was consistently available to participants.

For centres considering a similar programme, the key insight is that reproducibility depends less on access to hardware or benchmark codes, and far more on sustained investment in human resources. Allocating time for experienced engineers to serve as facilitators is essential to replicating the impact of the CHPC SCC.

9 CONCLUSION

For more than a decade, the CHPC Student Cluster Competition has reshaped HPC outreach and education in South Africa, demonstrating that rapid technical upskilling and international competitiveness are possible even in resource-constrained environments. By combining focused, resource-efficient training with collaborative, high-stakes competition, the programme has cultivated a generation of capable practitioners and innovators.

Beyond its national impact, the CHPC SCC presents a scalable model for HPC education worldwide. Its emphasis on technical rigor, teamwork, and inclusive opportunity offers a proven framework that other regions can adapt to build their own pipelines of HPC talent and strengthen participation in the global research community.

ACKNOWLEDGMENTS

We gratefully acknowledge the foundational contributions of Sakhile Masoka, and Israel Tsililo, whose dedication and vision were instrumental in the early successes of this programme.

We also extend our sincere thanks to the many staff members who contributed their expertise to the development of lecture content, tutorials, and technical support. Their commitment to excellence and education has been vital to the programme's growth.

A special tribute goes to all the student cluster teams and mentors who have participated over the years. Their enthusiasm, ingenuity, and perseverance have not only driven the programme forward but have also embodied its core mission of inclusive, high-performance computing education across the continent.

We further thank TACC and Dell Technologies for their generous assistance, and ISC-HPC for consistently hosting student cluster competitions that inspire and elevate emerging talent.

Some limited editing of grammar and phrasing was performed using Microsoft CoPilot and ChatGPT.

REFERENCES

[1] Michael Beukman, Sayfullah Jumoorty, Jonathan Luke Faller, Eugene de Beste, and Bryan Johnston. 2024. Second Life: Enabling a Student Special Interest Group through Decommissioned HPC Systems. In *Practice and Experience in Advanced Research Computing (PEARC '24)*. Association for Computing Machinery, New York, NY, USA, Article 92, 4 pages. https://doi.org/10.1145/3626203.3670573

[2] CSIR. n.d.. OpenNebula KVM SR-IOV driver. https://researchspace.csir.co.za/items/9385e27c-2082-4ba9-b386-26791a54eb99.

[3] Bryan Johnston. 2019. HPC Ecosystems Project: Facilitating Advanced Research Computing in Africa. In *Practice and Experience in Advanced Research Computing (PEARC '19)*. Association for Computing Machinery, New York, NY, USA, Article 107, 3 pages. https://doi.org/10.1145/3332186.3333264

[4] Bryan Johnston, Lara Timm, Eugene de Beste, and Mabatho Hashatsi. 2024. Let's Get Our Heads Out of the Clouds: A scalable and sustainable approach to HPC Training Labs for Resource Constrained Environments and anyone else stuck in the Clouds. *Journal of Computational Science Education* 15, 1 (March 2024), 64–71. https://doi.org/10.22369/issn.2153-4136/15/1/13 Presented at Supercomputing SC23.

[5] Bryan Johnston, Lara Timm, David Macleod, and John Poole. 2024. Ten Years of the HPC Ecosystems Project: Transforming HPC in Africa for the Past Decade. In *Practice and Experience in Advanced Research Computing (PEARC '24)*. ACM, 1–11. https://doi.org/10.1145/3626203.3670537

[6] Bryan Johnston, Lara Timm, David Macleod, John Poole, Lily de Melo, Sayfullah Jumoorty, Jonathan Faller, Reinhard Jansen van Vuuren, and Michael Beukman. 2025. From Student SIG to Success: The journey of a student HPC Special Interest Group towards sustainable training and success in Student Cluster Competitions. *Journal of Computational Science Education* 16, 1 (March 2025), 43–49. https://doi.org/10.22369/issn.2153-4136/16/1/9

[7] Promise Mvelase, Nomusa Dlodlo, David Macleod, and Happy Sithole. n.d.. *The Building Blocks to the Architecture of a Cloud Platform.* Technical Report. Council for Scientific and Industrial Research (CSIR).

[8] Dan Olds. 2019. ISC 2019 Student Cluster Competition: Day-by-Day Drama, Winners Revealed. https://insidehpc.com/2019/07/isc-2019-student-cluster-competition-day-by-day-drama-winners-revealed/. *InsideHPC* (2019). Accessed: 2025-09-09.

[9] South African Government. 2018. Science and Technology: International Student Cluster Competition. https://www.gov.za/news/media-statements/science-and-technology-international-student-cluster-competition-11-dec-2018. Intel Best Female Student Award introduced at the 2018 CHPC Student Cluster Competition, awarded to Mapule Madzena (University of the Free State).

Teaching AI Through Narrative Data: A Practical Framework for Data Science and Retrieval-Augmented Generation

Charlie Dey
Texas Advanced Computing Center
charlie@tacc.utexas.edu

Susan Lindsey
Texas Advanced Computing Center
slindsey@tacc.utexas.edu

ABSTRACT

Artificial intelligence (AI) and machine learning (ML) education has traditionally been split between technical model-building and data literacy. While these skills are often taught separately, the emergence of large language models (LLMs) offers an opportunity to unify them through narrative-driven, human-readable data transformation. This approach enables learners to query structured data using natural language while still engaging deeply with the underlying analytical processes.

We present a hands-on educational framework—debuting at the 2025 Big Data School in Costa Rica—that grounds AI learning in real-world data by transforming a single, richly structured dataset into narrative text that LLMs can ingest and reason over. Using the Austin Real-Time Traffic Incident Reports, participants apply core data science techniques—classification, clustering, regression, and forecasting—before extending their work into Retrieval-Augmented Generation (RAG) pipelines. A key design element is the careful selection of a dataset that supports multiple analytical tasks, contains time-series and categorical diversity, and can be effectively compressed into natural-language summaries without losing critical meaning. This continuity of dataset enhances comprehension, data intuition, and knowledge transfer without the cognitive effort of switching contexts.

By converting tabular data into concise, human-readable narratives, learners bridge traditional analytics and AI-enhanced insight generation. Outcomes include skill-building in Python-based ML pipelines, embedding and vector retrieval, and critical reflection on model interpretability, hallucination risk, and accessibility for non-technical users. This narrative-driven, single-dataset strategy—combined with explicit dataset selection principles—supports scalable adoption for high-performance computing and AI pedagogy in both academic and professional environments.

KEYWORDS

AI Education, Narrative Data, Retrieval-augmented Generation, Machine Learning Pedagogy, Data Storytelling

1 INTRODUCTION

Teaching artificial intelligence is no longer just about building models or tuning parameters — it's about helping people connect data, algorithms, and real-world meaning. In many AI and machine learning courses, these skills are taught in isolation: students learn data cleaning in one place, model building in another, and never see how the pieces fit together in practice. Large language models (LLMs) change that equation. By combining analytical rigor with the flexibility of natural language, they give us new ways to explore and explain data.

But there's a catch: most datasets used in teaching aren't built with this kind of integration in mind. They're either too small to show the complexity of real data, too large to work with interactively, or too fragmented to connect multiple techniques in a single flow. The result? Learners spend more time switching datasets and contexts than building a deep understanding of the data itself.

This work proposes a different approach: choose one rich, well-structured dataset and use it as the backbone for an entire AI learning journey. By converting that dataset into narrative form — short, human-readable stories that LLMs can reason over — we make it possible to bridge traditional analytics with LLM-powered exploration. The result is a smoother learning experience, where concepts build naturally on each other and learners see AI as both a set of tools and a way of thinking.

We present a framework for teaching AI that centers on a single, real-world dataset. Using the Austin Real-Time Traffic Incident Reports [2], participants explore the dataset through a series of ML tasks before transforming the data into narrative form for ingestion by LLMs. This supports a pedagogical model where learners move fluidly from traditional analytics to AI-enhanced question answering.

2 BACKGROUND AND RELATED WORK

AI and machine learning (AI/ML) education has expanded rapidly in recent years, driven by the growth of online courses, workshops, and bootcamps. While these initiatives have broadened access, they often share a common limitation: fragmentation. Learners are frequently exposed to isolated exercises tied to disparate datasets—a small CSV for cleaning in one session, a regression model on a different dataset in the next, and a visualization task on yet another. Each skill is valuable on its own, but this "patchwork" approach makes it difficult for students to see how the pieces of the data science pipeline—acquisition, cleaning, feature engineering, modeling, interpretation, and communication—fit together in practice.

Research underscores the importance of continuity in overcoming this barrier. Segel and Heer [7] emphasize that storytelling helps sustain engagement and comprehension in data-driven contexts, while Chen et al. [1] show that embedding narrative elements into data science courses strengthens the link between technical proficiency and conceptual understanding. Yet most AI/ML curricula

https://doi.org/10.22369/issn.2153-4136/17/1/7

remain disconnected, lacking a unifying dataset or sustained problem context that can scaffold learning across modules. As O'Reilly et al. [5] note, such fragmentation increases the cognitive burden on learners, who must continually reset their mental models as they move between unrelated domains and data structures.

This challenge is not unique to AI/ML. Huppenkothen et al. [4], in their work on hack weeks, highlight the benefits of sustained, project-based learning environments where participants engage with the full lifecycle of data-intensive research. Similarly, Powell et al. [6] demonstrate that hackathon-style approaches foster workforce development by immersing learners in end-to-end problem solving. Both approaches emphasize continuity, collaboration, and problem ownership—qualities often missing from short, fragmented workshop formats.

Our framework builds on these insights. By centering instruction around a single, complex dataset—the Austin Real-Time Traffic Incident Reports—we provide a continuous narrative arc that spans multiple instructional modules. This design allows students to progress from cleaning and feature engineering to classification, clustering, regression, and forecasting, all within the same context. Rather than treating algorithms as isolated tools, learners see how research questions, feature choices, and modeling decisions connect within a unified workflow that mirrors authentic AI/ML practice.

Narrative visualization has been recognized as a powerful means of communicating data and enhancing learner engagement [1, 7]. Embedding storytelling into analysis not only fosters clarity but also helps learners develop a deeper intuition for data. However, existing approaches rarely integrate ML training, narrative data transformation, and retrieval-based exploration into a single cohesive framework. Teaching with large datasets introduces further challenges of reproducibility, scalability, and accessibility [5]. Collaborative models like hack weeks [4] mitigate some of these issues, but few extend into modern AI workflows.

To situate our contribution within a broader pedagogical context, this framework aligns with principles of Project-Based Learning (PBL). PBL emphasizes sustained inquiry, real-world problem contexts, and iterative exploration, helping students develop both technical mastery and metacognitive skills. By anchoring the curriculum in a single, richly structured dataset, we extend PBL principles into the AI/ML domain: learners engage with the full lifecycle of data science, ask their own research questions, experiment with algorithmic choices, and reflect on successes and failures. This provides not only technical proficiency but also a holistic understanding of how data, methods, and narratives intersect in authentic practice.

3 METHODOLOGY

Our methodology is designed to create a cohesive, end-to-end learning experience that mirrors the lifecycle of a real AI/ML project. Rather than presenting algorithms in isolation, we anchor the workshop around a single, complex dataset that can sustain multiple analytical techniques and narrative perspectives. This design reduces the cognitive overhead of repeatedly switching between datasets, while allowing participants to progressively build intuition, technical proficiency, and storytelling skills in a unified context.

The methodology unfolds in four stages. First, we establish dataset selection principles to ensure that the chosen data can support diverse forms of analysis while remaining interpretable to learners. Next, participants engage in data familiarization, a stage focused on developing intuition about the dataset's structure, distributions, and potential features. Once a foundation of familiarity is established, the data is subjected to narrative transformation, where structured records are recast into natural language statements that remain semantically stable but more accessible for reasoning and retrieval tasks. Finally, participants integrate these narrative records into AI/ML and RAG workflows, combining traditional analytical techniques with retrieval-augmented generation pipelines.

Together, these stages create a scaffolded yet flexible framework that supports both technical mastery and narrative coherence, preparing learners to move fluidly between coding, analysis, and communication.

3.1 Dataset Selection Principles

Selecting the right dataset is central to the framework. In contrast to traditional workshops that rotate through small, disconnected datasets, we sought a dataset that could sustain **multiple analytical perspectives** and provide learners with opportunities to practice aligning research questions, features, and models. Our criteria for dataset selection were:

- **Enable multiple analytical techniques.** The dataset needed to naturally support a range of methods including classification, clustering, regression, and time-series forecasting. This diversity allows participants to experience how different models answer different types of questions, such as "What category does this belong to?" versus "What value will occur next?"
- **Contain temporal, categorical, and textual elements.** A variety of data types ensures that learners encounter the practical challenges of feature engineering, encoding, and preprocessing. For example, timestamps can drive forecasting models, categorical labels inform classification, and textual notes provide opportunities for embedding-based retrieval.
- **Balance complexity and accessibility.** The dataset must be large and realistic enough to feel "authentic," but not so large that it overwhelms learners during interactive exploration. This balance supports hands-on work while preventing the workshop from devolving into a purely computational challenge.
- **Be semantically stable under narrative transformation.** Because a key component of the framework is converting structured records into narrative form, the data had to retain its essential meaning whether represented as a table, a feature set, or a natural language statement.
- **Offer real-world relevance.** Context matters for engagement. A dataset tied to issues like urban traffic and public safety resonates with learners while also supporting authentic analytical challenges.

Beyond these structural considerations, the dataset also needed to facilitate **model selection as a pedagogical activity**. One workshop module asked participants to identify potential research questions and then map those questions to suitable models. For example, predicting whether an incident is likely to involve injury calls for a classification model, while estimating the expected response time benefits from regression. Clustering can be used to discover hidden patterns in incident types, and time-series forecasting enables projections of traffic incidents by time of day or season. By grounding these choices in a single, multifaceted dataset, students were able to see how different models carry distinct assumptions and data requirements, and why choosing “the right tool for the job” is a critical skill in AI/ML practice.

3.2 Data Familiarization

Before applying any machine learning techniques, participants engaged in a dedicated stage of **data familiarization**. This phase was designed to build what we call “data intuition”—an understanding of not only the structure of the dataset, but also its limitations, potential, and the kinds of questions it can reasonably answer.

Participants began by examining the dataset’s schema, field definitions, and metadata. They explored distributions of numerical features, frequency counts of categorical variables, and patterns in temporal fields. For example, learners inspected how incidents varied by time of day, type of report, and location zone. This process was not simply descriptive but interpretive, as participants were encouraged to ask:

- What trends or anomalies stand out in the raw data?
- Which features seem relevant to the research questions we might ask?
- Are there missing values or inconsistencies that could bias the models?
- How might the temporal, categorical, or textual aspects of the data inform feature engineering?

Visual exploration played a central role. Using `pandas` for summary statistics and `matplotlib` for visualization, participants generated histograms, scatter plots, and heatmaps to observe relationships and outliers. These exercises helped learners become comfortable with the dataset before any modeling occurred.

The goal of this phase was twofold: (1) to reduce the cognitive burden of treating the dataset as a black box, and (2) to lay a foundation for later discussions about **model-data fit**. By first developing familiarity with the data, participants were better prepared to reason about which models would succeed, which would fail, and why. This aligns with project-based learning practices, where understanding the problem context is as important as technical implementation.

3.3 Narrative Transformation

Once participants had established a solid understanding of the dataset, the next step was to translate structured tabular records into short, human-readable narratives. This process, which we call **narrative transformation**, served two purposes: (1) to help learners consider how structured data can be expressed in natural language, and (2) to prepare the data for exploratory exercises with large language models (LLMs) later in the workshop.

For example, a traffic incident record originally stored as structured fields:

- Timestamp: 2025-08-14 17:43
- Location: Interstate 35, Zone 4
- Type: Collision
- Reporting Agency: Austin Police Department

could be transformed into a narrative record such as:

> “At 5:43 PM on August 14th, a collision occurred on Interstate 35 in zone 4 of Austin as reported by the Austin Police Department.”

This conversion preserved the semantic content of the original record while making it accessible to both human readers and natural language systems. Participants experimented with different levels of detail—from minimal fact-based sentences to context-rich descriptions—and reflected on how narrative framing influences interpretation.

Importantly, the narrative transformation exercise was not just a prelude to LLM use. It also provided a bridge back to traditional feature engineering by raising questions such as:

- Which fields are essential to preserve, and which can be omitted without loss of meaning?
- How do phrasing and word choice affect interpretability for humans versus automated systems?
- What risks of ambiguity or bias are introduced during narrative construction?

Rather than immediately building a full retrieval-augmented generation (RAG) system, the workshop used these narrative records to *explore* how LLMs handle structured-to-text conversions, embedding creation, and simple retrieval tasks. This positioned narrative transformation as a conceptual hinge between conventional machine learning workflows and emerging AI-assisted data exploration, without requiring participants to fully master production-grade RAG pipelines.

3.4 Integrated AI/ML and Exploratory RAG Workflows

The final phase of the workshop was designed to connect traditional machine learning pipelines with exploratory uses of large language models (LLMs). This integration unfolded in two stages.

3.4.1 End-to-End AI/ML Pipelines. Participants first completed the full sequence of steps in a conventional machine learning workflow:

- **Data preparation** — Cleaning, normalizing, and splitting the traffic incident dataset into training and testing sets.
- **Feature engineering** — Identifying relevant features such as time of day, incident type, or location zone, and creating derived variables (e.g., weekday vs. weekend, rush-hour indicators).
- **Model exploration** — Matching research questions to algorithms, such as:
  - *Classification:* Can we predict whether an incident will require emergency response?
  - *Clustering:* What natural groupings of incidents emerge from location and severity?
  - *Regression:* How do weather or time features correlate with incident duration?
  - *Forecasting:* Can we project incident frequency trends across future weeks?

- **Model selection and tuning** — Comparing performance metrics, discussing overfitting risks, and iteratively refining hyperparameters.

This phase emphasized not only running models but also *choosing the right model for the right question.* By explicitly mapping data requirements (e.g., categorical vs. continuous features, time-dependence) to algorithm capabilities, students developed intuition for how methodological choices shape research outcomes [3].

3.4.2 Exploratory RAG Extensions. Once participants had completed the ML pipelines, they returned to their narrative-transformed records to explore how LLMs could add value. Rather than constructing a full production retrieval-augmented generation (RAG) pipeline, the workshop focused on lightweight experiments:

- Creating embeddings of narrative records and storing them in a vector database.
- Querying the vector database with natural language prompts (e.g., "Find recent collisions near Interstate 35 during rush hour").
- Comparing how well the retrieval results aligned with answers generated through traditional SQL or Pandas queries.
- Reflecting on issues of precision, ambiguity, and hallucination when using LLMs for data-grounded tasks.

This exploratory approach allowed students to see how narrative data transformations could be leveraged by modern AI tools, while avoiding the complexity of building full-scale RAG systems. In doing so, the workshop highlighted the complementary strengths of structured-query methods and LLM-based retrieval, preparing learners to critically assess when each is appropriate.

3.4.3 Hackathon-Style Application. To reinforce these skills, the workshop culminated in a hackathon-style exercise. Student teams selected their own datasets or worked with subsets of the traffic incident dataset, then:

- Defined three research questions of interest.
- Identified what features would be necessary to address those questions, including possible engineered variables.
- Mapped their questions to candidate algorithms, explicitly discussing which models could succeed or fail and why.
- Made assumptions about their data (e.g., independence, completeness, or stationarity) and presented these in team introductions.
- Implemented, tuned, and compared algorithms in practice.
- Presented their findings, reflecting on both successful and unsuccessful approaches.

This design encouraged participants to approach machine learning as an open-ended process of inquiry rather than a fixed recipe. The hackathon not only reinforced technical content but also fostered collaboration, problem framing, and critical thinking. We expand on the outcomes of this exercise in Section 3.5.

3.4.4 Exploratory RAG Extensions. Once participants had completed the ML pipelines, they returned to their narrative-transformed records to explore how LLMs could add value. Rather than constructing a full production retrieval-augmented generation (RAG) pipeline, the workshop focused on lightweight experiments:

- Creating embeddings of narrative records and storing them in a vector database.
- Querying the vector database with natural language prompts (e.g., "Find recent collisions near Interstate 35 during rush hour").
- Comparing how well the retrieval results aligned with answers generated through traditional SQL or Pandas queries.
- Reflecting on issues of precision, ambiguity, and hallucination when using LLMs for data-grounded tasks.

This exploratory approach allowed students to see how narrative data transformations could be leveraged by modern AI tools, while avoiding the complexity of building full-scale RAG systems. In doing so, the workshop highlighted the complementary strengths of structured-query methods and LLM-based retrieval, preparing learners to critically assess when each is appropriate.

3.5 Hackathon Outcomes

The hackathon-style exercise served as the culminating activity of the workshop, providing participants with an opportunity to synthesize technical content, methodological reasoning, and collaborative problem-solving. While prior sessions emphasized guided exploration of specific techniques, the hackathon placed students in a semi-open environment where they were responsible for framing their own research questions, selecting analytical approaches, and presenting results.

3.5.1 Team Problem Framing. Each team began by selecting a dataset (either a subset of the traffic incident data or an alternative dataset of personal interest) and proposing three guiding research questions. Examples included:

- *Classification:* Can we predict whether an incident requires a police response based on time, location, and type?
- *Regression:* What factors are most predictive of incident duration?
- *Forecasting:* Can we estimate how traffic incidents will evolve during upcoming weeks?

Teams were required to articulate the assumptions underlying their questions (e.g., data completeness, independence of samples, or stationarity in time series). This step highlighted the importance of transparency in computational science, as assumptions shape both the feasibility and interpretation of results.

3.5.2 Model Development and Experimentation. Once questions were defined, teams mapped them to candidate machine learning methods. For instance, classification problems often involved decision trees or logistic regression, while clustering tasks employed k-means or hierarchical approaches. Teams were encouraged to experiment with multiple algorithms, compare performance, and reflect on trade-offs such as interpretability versus predictive accuracy.

By aligning models to questions, participants gained a deeper appreciation of the iterative nature of AI workflows: each decision about features, algorithms, and tuning parameters reflected a hypothesis about the underlying data.

3.5.3 Presentations and Peer Review. Hackathon teams concluded with short presentations to the broader workshop group. Presentations included:

- A clear statement of research questions and motivating rationale.
- The dataset and features selected, including engineered variables.
- The algorithms applied, performance comparisons, and tuning efforts.
- A discussion of assumptions and potential limitations.
- Reflections on what worked, what failed, and how results could inform further inquiry.

Peer participants were invited to ask questions, challenge assumptions, and suggest alternative modeling strategies. This peer review element emphasized scientific communication and constructive critique as essential components of computational practice.

3.5.4 Observed Educational Impact. The hackathon outcomes demonstrated several key gains for participants:

- **Critical thinking:** Students learned to evaluate not only how models perform, but also whether they are appropriate for the question at hand.
- **Collaboration:** Teams reported that framing problems and dividing tasks required negotiation and clear communication.
- **Adaptability:** Many groups encountered data limitations or unexpected model behaviors, requiring them to pivot strategies.
- **Confidence:** By presenting to peers, participants demonstrated increased comfort discussing both successes and failures in technical work.

Collectively, these outcomes illustrate how hackathon-style integration can transform abstract machine learning concepts into lived experience. Rather than following step-by-step tutorials, participants engaged with the uncertainties and trade-offs that define real-world computational research.

4 OUTCOMES

The outcomes of this workshop can be considered along three dimensions: participant learning gains, the development of an integrated RAG-enabled workflow, and the collaborative hackathon that culminated the experience. Together, these outcomes provide evidence that the framework effectively bridges fragmented AI/ML training with a unified, project-based learning experience.

4.1 Learning Gains

Participants demonstrated measurable improvement in their ability to connect individual analytical tasks to the broader data science lifecycle. Early exercises revealed that many students could apply standard ML algorithms (e.g., linear regression or k-means) in isolation, but struggled to articulate how such methods mapped to different kinds of research questions. By the end of the workshop, participants showed stronger competency in aligning questions with appropriate methods, understanding data requirements for specific models, and communicating results in both technical and narrative forms. Informal assessments and group discussions suggested that this progression was due to the continuity of working with a single dataset throughout the workshop.

4.2 Integrated RAG Workflow

Another important outcome was the successful incorporation of a retrieval-augmented generation (RAG) pipeline. Instead of presenting RAG as an isolated “add-on,” the workshop positioned it as a natural extension of the narrative transformation process described in Section 3. This allowed participants to experience how large language models (LLMs) can query, summarize, and contextualize data when supported by an external knowledge base. Embedding RAG within the workflow highlighted both its potential and its limitations: students noted that while the approach excelled in exploratory analysis and generating human-readable summaries, it required careful tuning and filtering to avoid hallucinations or irrelevant responses. These reflections helped frame RAG as a complementary, not replacement, technology within applied ML.

4.3 Collaborative Hackathon

The capstone hackathon provided the most visible demonstration of learning outcomes. In small teams, participants were tasked with selecting a subset of the traffic dataset and framing three distinct research questions. Teams had to justify why their chosen questions were both meaningful and feasible, identify the features necessary to answer them, and propose algorithms that could address those questions effectively. This process required teams to articulate assumptions and trade-offs, mirroring authentic data science practice.

During the hackathon, participants iteratively refined their approaches by cleaning data, engineering features, and experimenting with supervised and unsupervised models. Teams also reviewed their assumptions when the results did not align with expectations, reinforcing the importance of critical evaluation in ML workflows. The final presentations included not only algorithmic results but also narrative justifications of the methodological choices. This underscored the dual focus of the workshop: technical competence and the ability to communicate insights clearly. Feedback from participants indicated that the hackathon was the most engaging part of the workshop, providing a strong sense of ownership and achievement.

4.4 Collaborative Hackathon (Expanded)

The capstone hackathon provided a structured yet open-ended environment for participants to apply the skills they had acquired throughout the workshop. Each team was responsible for selecting a subset of the traffic dataset and developing three research questions that were both analytically tractable and pedagogically meaningful. Teams needed to evaluate what features would be required to address their questions effectively and determine which algorithms would best answer them. This process involved considerations such as data sparsity, feature types, temporal dependencies, and model assumptions, reflecting the types of decisions data scientists make in professional contexts.

Teams then presented their preliminary approaches, assumptions, and anticipated challenges during a "team introduction" session. This encouraged peer feedback and fostered a culture of collaborative problem solving. Throughout the hackathon, participants iteratively refined their models, conducted feature engineering, and compared performance across algorithms. Failures were treated as learning opportunities: when models underperformed or produced unexpected results, students revisited assumptions, discussed potential data issues, and considered alternative analytical strategies.

Final team presentations emphasized both the technical outcomes and the narrative framing of their work. Students demonstrated competence in applying classification algorithms, clustering methods, regression models, and forecasting approaches within a single dataset, illustrating how the workshop's design facilitated knowledge transfer across techniques. The hackathon experience highlighted critical aspects of AI/ML education: linking questions to methods, evaluating model suitability, and communicating findings effectively.

4.5 Limitations and Reflections

While the single-dataset approach proved effective for building continuity and cohesion, it also imposed certain constraints. The choice of dataset determined the scope of research questions and analytical techniques that could be explored. Some participants noted that a more diverse set of datasets might allow exploration of additional model types or data modalities. Additionally, time constraints limited the depth to which participants could explore complex models or advanced feature engineering techniques.

Despite these limitations, feedback suggested that the integrated, project-based format enhanced participant confidence and engagement. The emphasis on asking the right questions, mapping them to appropriate algorithms, and critically reflecting on model assumptions provided a holistic understanding of the AI/ML pipeline. Participants also reported increased appreciation for the challenges inherent to real-world data and the value of iterative, exploratory learning in mastering AI/ML workflows. By structuring the workshop around a single, richly detailed dataset, learners experienced continuity across activities that is rarely seen in traditional AI/ML trainings. The Project-Based Learning (PBL) approach allowed participants to formulate research questions, explore feature requirements, and select models appropriate to the data and questions at hand. As illustrated in Table **??**, each activity built on the previous one, reinforcing key skills and concepts while maintaining a coherent narrative thread. This continuity not only reduced cognitive load typically associated with switching datasets and contexts but also enabled students to observe how decisions in data preprocessing, model selection, and evaluation interconnect in real-world workflows. The hackathon portion further strengthened these outcomes by letting learners independently apply the same pipeline on novel datasets, fostering critical thinking, problem-solving, and reflection on algorithm suitability and data quality.

5 NOVELTY AND IMPACT

This structured, continuous learning experience not only helped participants gain technical skills but also reinforced critical thinking and problem-solving across the full AI/ML workflow. By guiding learners from data familiarization through feature engineering, model selection, and evaluation within a single dataset, the workshop fostered a deeper understanding of algorithmic choices and data constraints. These outcomes naturally lead into the broader discussion of the pedagogical novelty and impact of this approach, highlighting how integrating project-based learning with a continuous dataset framework differentiates this workshop from more fragmented AI/ML training models. This workshop framework differs from most AI/ML educational offerings in several key ways:

- **Single-dataset integration across the curriculum:** By centering instruction on a single, rich dataset—the Austin Real-Time Traffic Incident Reports—participants experienced continuity across multiple analytical tasks, from data cleaning to feature engineering, classification, clustering, regression, and forecasting. This approach reduced the cognitive effort typically associated with switching datasets, helping learners build deeper data intuition and a cohesive understanding of the AI/ML pipeline.
- **Project-based, question-driven learning:** Incorporating a capstone hackathon emphasized asking the right questions, selecting appropriate models, and iteratively refining solutions. This mirrors real-world data science workflows and leverages learning through failure, encouraging critical thinking, problem ownership, and collaborative skill-building.
- **Bridging analytical techniques with model selection principles:** Students were explicitly guided to consider which algorithms were best suited to their research questions and dataset characteristics. This included discussions on data criteria (e.g., feature types, sparsity, temporal dependencies), algorithm assumptions, and the interpretability of model outputs—topics often overlooked in conventional workshops.
- **Reproducibility and scalability:** All code and workflows were implemented in Python using open-source libraries, ensuring that the framework can be adapted for other datasets, institutions, or professional settings. By providing detailed instructional materials, the workshop design supports reproducible and scalable AI/ML pedagogy.
- **Holistic pedagogical impact:** Unlike traditional workshops that isolate skills, this integrated approach helps learners develop a broad understanding of the end-to-end AI/ML process. Participants reported increased confidence in applying methods across different tasks, linking questions to algorithms, and communicating findings—skills directly relevant to both academic and industry contexts.

Taken together, these elements contribute to a novel educational paradigm that emphasizes continuity, practical skill-building, and reflective learning. By combining project-based instruction with single-dataset integration, this framework addresses a documented gap in AI/ML education: the fragmentation of learning experiences across multiple, disconnected datasets and exercises. This paper provides a model that other educators can replicate, adapt, and expand upon in diverse instructional settings.

Table 1: Workshop Outcomes: Learning Objectives, Activities, and Skills Developed

Learning Objective	Workshop Activity	Skills Developed
Data Familiarization	Exploring the Austin Traffic Incident dataset, examining distributions, spotting anomalies, visualizations	Data intuition, EDA, feature identification
Data Cleaning and Feature Engineering	Handling missing values, encoding categorical variables, feature scaling	Python/Pandas, preprocessing pipelines, reproducible workflows
Classification	Logistic regression, decision trees; discussion of entropy, information gain, CountVectorizer/TfidfVectorizer	Model understanding, feature importance, algorithm selection, analytical reasoning
Clustering	K-Means, DBSCAN; evaluating cluster quality	Unsupervised learning, parameter tuning, pattern discovery
Regression	Linear regression, Random Forest, revisiting overfitting/underfitting	Predictive modeling, model evaluation, error analysis
Time-Series Forecasting	ARIMA; trend and seasonality analysis	Forecasting, temporal reasoning, model selection criteria
Hackathon Project	Teams choose a dataset, formulate 3 research questions, determine needed features and algorithms, run experiments, present results	Problem formulation, dataset-to-question mapping, experimental design, teamwork, presentation skills

6 DISCUSSION AND FUTURE WORK

The workshop described in this paper demonstrates a practical, integrated approach to AI/ML education, yet there are several areas for refinement and expansion. One key limitation is the scope of assessment: while qualitative feedback and student reflections indicate enhanced comprehension and confidence, future iterations will incorporate structured evaluation metrics to measure learning gains, algorithmic understanding, and data literacy improvements. For example, pre- and post-workshop assessments, alongside rubrics for hackathon projects, could provide quantitative evidence of skill acquisition.

Reproducibility and scalability are also central considerations. All materials were developed using open-source Python libraries and documented workflows, enabling replication at other institutions. However, wider deployment may require adaptation for different datasets, class sizes, or learner backgrounds. Future work will explore modularizing the framework so that instructors can plug in alternative datasets while maintaining continuity and coherence across tasks.

Expanding the workshop series will also provide the opportunity to collect longitudinal data on learning outcomes, model selection proficiency, and project-based reasoning. Tracking how learners transfer skills to new datasets or real-world problems will help validate the framework's effectiveness beyond the initial pilot. Additionally, integrating more explicit discussions of ethical considerations, bias in models, and interpretability trade-offs will strengthen students' understanding of responsible AI practice.

Finally, future iterations could examine hybrid or fully remote implementations, leveraging cloud-based platforms or high-performance computing resources to broaden accessibility. By iterating on these dimensions, we aim to create a replicable, evidence-based model for comprehensive AI/ML instruction that addresses fragmentation, enhances learner engagement, and prepares participants for authentic data-driven challenges.

ACKNOWLEDGEMENTS

We thank the organizers of the 2025 Big Data School in Costa Rica for the opportunity to pilot this workshop. We also thank fellow TACCsters Dr. Joe Stubbs and Dr. Anagha Jamthe for their work in developing an extensive course curriculum on AI/ML for the Computational Engineering department at the University of Texas, Austin [8].

REFERENCES

[1] Lujie Karen Chen, Jamie Gillan, Matthew Decker, Egan Eteffa, Anjelica Marzan, Justin Thai, and Sarah Jewett. 2023. Embedding Digital Data Storytelling in Introductory Data Science Course. *Journal of Problem Based Learning in Higher Education* 11, 2 (2023).

[2] City of Austin Open Data. 2025. Real-Time Traffic Incident Reports. https://data.austintexas.gov/Transportation-and-Mobility/Real-Time-Traffic-Incident-Reports/dx9v-zd7x. Accessed: 2025-9.

[3] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning*. Springer.

[4] Daniela Huppenkothen, Anthony Arendt, David W. Hogg, Karthik Ram, Jacob T. VanderPlas, and Ariel Rokem. 2017. Hack weeks as a model for data science education and collaboration. *ArXiv preprint arXiv:1711.00028* (2017).

[5] Catherine M. O'Reilly, Tanya Josek, Rebekka D. Darner, and Sarah K. Fortner. 2022. Pedagogy of teaching with large datasets. *Biochemistry and Molecular Biology Education* 50, 5 (2022), 439–447.

[6] Jeaime Powell, Linda Bailey Hayden, Amy Cannon, John Holly, Charlie Dey, and Alexander Nolte. 2022. The HackHPC Model: Fostering workforce development in high-performance computing through hackathons. *Gateways 2022 Proceedings*. https://par.nsf.gov/biblio/10450534

[7] Edward Segel and Jeffrey Heer. 2010. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1139–1148.

[8] Joe Stubbs and Anagha Jamthe. 2025. COE 379L: Software Design for Responsible Intelligent Systems, Spring 2025. https://coe379l-sp25.readthedocs.io/en/latest/. Course materials and syllabus for a machine learning and data systems class at The University of Texas at Austin.

Enhancing HPC Curriculum through Competitions

Cristina Carbunaru
National University Of Singapore
cristina@nus.edu.sg

Sriram Sami
National University Of Singapore
sriramsami@nus.edu.sg

ABSTRACT

High Performance Computing (HPC) supports breakthroughs in artificial intelligence (AI), data-intensive science, and engineering. At the National University of Singapore (NUS), core parallelism concepts are currently taught through courses in Parallel Computing and Concurrent Programming, with additional domain-specific exposure in courses. While these offerings build strong theoretical foundations, they leave a gap in systems-level competencies essential for deploying, optimizing, and scaling applications on real HPC infrastructure.

We addressed this gap by initiating several projects meant to increase the knowledge in system-level skills for HPC. A main initiative is the participation in HPC student cluster competitions through which we integrated training in resource management, profiling, monitoring, containerized workflows, and distributed AI workloads for our selected students. This focus enables participants to bridge programming theory with operational expertise, preparing them to work effectively with clusters and heterogeneous architectures. Building on the interest around HPC competitions, the main curriculum in computer science is developing to include full-fledged HPC courses.

We faced several challenges in this process, including a steep learning curve with complex systems, limited access to costly and shared cluster resources, and a shortage of instructors with up-to-date expertise. Pedagogically, bridging theory and large-scale practice is difficult, especially in the HPC context where the access to resources is remote. Therefore, sustainable curriculum development calls for a gradual expansion of teaching topics and resources, coupled with the integration of hands-on, competition-driven learning to maintain engagement.

Formal HPC training enhances students' readiness for careers in computational science, promotes cross-disciplinary collaboration, and equips graduates with the advanced skills essential for solving complex challenges in AI and data-intensive fields.

KEYWORDS

Student Competitions, Artificial Intelligence, HPC Education

1 INTRODUCTION

With the increasing interest in high-performance computing generated by the need to manage and harvest computation resources for large scientific tasks and AI, we embarked on a journey to improve the HPC curriculum at the National University of Singapore (NUS). Foundational parallelism concepts are taught through courses such as CS3210 Parallel Computing, CS3211 Parallel and Concurrent Programming, providing strong theoretical grounding and programming skills, but they do not fully address the systems-level competencies required for deploying, optimizing, and scaling workloads on real HPC infrastructure.

This paper presents the existing courses related to HPC and illustrates the efforts of enhancing HPC training and education at NUS. We adopted an incremental approach by integrating hands-on training through final year projects (FYPs) and initiating participation in international student cluster competitions. These activities expose students to advanced topics such as resource management, distributed AI workloads, performance profiling, monitoring, and containerized workflows, which are skills essential for managing heterogeneous HPC clusters. The growing interest in HPC has motivated the proposal for a dedicated HPC course [1].

2 EXISTING PARALLEL COMPUTING COURSES

CS3210, CS3211, and CS5239 together form a strong foundation for HPC education at NUS by covering parallel architectures, programming paradigms, and performance optimization – key pillars in large-scale computing [2]. CS3210 Parallel Computing focuses on the theory and practice of parallelism, exposing students to shared-memory, distributed-memory, GPU, and heterogeneous systems. It teaches parallel computation models, algorithm design, scalability analysis, and performance measurement – skills directly applicable to HPC cluster workloads. CS3211 Parallel and Concurrent Programming deepens this by exploring concurrency in modern languages such as C++20, Go, and Rust. It addresses synchronization, safety, robustness, and performance trade-offs, enabling students to write efficient, scalable, and safe parallel software for emerging technologies and HPC environments. CS5239 Computer System Performance Analysis complements both by teaching latency, utilization, bottleneck analysis, and queuing theory. Its focus on measurement, workload characterization, and tuning mirrors the profiling and optimization work essential in HPC system and application performance engineering. Together with other related courses, such as Distributed Systems and Cloud Computing, they provide the theoretical, programming, and performance analysis skills needed for advanced HPC roles.

3 INITIATING SYSTEM ADMINISTRATION (DEVOPS) TRAINING

Managing technical resources for courses such as CS3210 Parallel Computing, CS3211 Parallel and Concurrent Programming, and CS5239 Computer System Performance Analysis presents notable challenges in ensuring effective use of the Parallel and Distributed

https://doi.org/10.22369/issn.2153-4136/17/1/8

Computing Lab cluster by about 400 students per semester, given its limited 24-node capacity. These challenges present rich opportunities for Final Year Projects (FYPs), where students can design and implement enhancements such as highly available, customizable monitoring infrastructures and automated software installation systems. We started training students on deployment, setting up, and monitoring technologies by engaging them in these undergraduate FYPs. These projects saw high interest from our students, which proved the interest in systems-level competencies beyond what the current curriculum could offer.

We proposed and developed a platform for managing the nodes for students and teaching staff. We created a lightweight and robust system that is customizable to support different course requirements, but at the same time, is easily shared by multiple groups of students and teaching staff. Trainees explored architecture design, software stack selection, integration difficulties, and potential operational issues, while their deployment and testing in real teaching environments would directly improve the educational experience of students. Students were trained on using tools such as Ansible (for deployment), Slurm (for job management), LDAP (for user management), Prometheus and Grafana (for monitoring and alerting). Mastering these tools equips students to manage HPC clusters efficiently – automating deployments, scheduling jobs, controlling user access, and monitoring performance in real time are skills essential for reliable, scalable, and secure operation of large-scale scientific and AI computing environments.

4 PARTICIPATION IN STUDENT CLUSTER COMPETITIONS

Participation in student cluster competitions directly enriches HPC curricula by providing hands-on, real-world experience that extends beyond classroom learning [3]. Students must design, configure, and optimize an actual HPC cluster under strict time and power constraints, applying concepts from parallel programming, systems performance, and resource management taught in courses like CS3210, CS3211, and CS5239.

Building on the training experience we gained through FYPs, we initiated the first NUS participation in Student Cluster Competitions in 2024. Our first NUS team, "Kent Ridge", achieved immediate and notable success – winning the Hero Run (HPLinpack) at IndySCC@SC24 in Atlanta and securing third place in the Virtual SCC@ISC25 in Hamburg. These results showcase the technical and collaborative capabilities fostered through competition training.

The overarching goal is to build a platform where students can deepen and showcase their expertise in HPC applications, hardware, and software while fostering collaboration, innovation, and networking within the global HPC community. So far, approximately 20 students have been trained on advanced HPC topics not currently covered in the standard NUS School of Computing's curricula, including cluster architecture, job scheduling, performance tuning, and advanced monitoring – bridging the gap between academic theory and professional HPC practice.

5 PROPOSING A HPC COURSE

The increasing interest in High Performance Computing (HPC) among students, generated by participation in cluster competitions and advanced project work, has motivated us to propose a dedicated HPC course at NUS. This course would go beyond the current parallel computing and performance analysis modules, providing students with hands-on experience in building, optimizing, and running applications on large-scale computing systems [4]. The course would focus on the integration of HPC with AI, including the design and deployment of distributed training infrastructure using frameworks such as Colossal-AI, and the development of efficient pipelines for large-scale model training. Students would also explore domain-specific HPC workflows in areas such as climate modeling, bioinformatics, and precision medicine, gaining exposure to real-world, data-intensive scientific challenges. The course would teach profiling and optimization techniques for large scientific applications, enabling students to identify performance bottlenecks and improve scalability across heterogeneous architectures. Practical work would leverage national and institutional HPC infrastructure, including the National Supercomputing Centre (NSCC) of Singapore and NUS IT resources, providing authentic experience with production-scale systems.

6 CONCLUSION

Empirical data collected from discussions with students and faculty shows a strong demand for enhanced HPC education at NUS. There is growing interest among students eager to deepen their knowledge in HPC technologies, coupled with a cohort of highly capable individuals ready to engage with advanced topics. This feedback reinforces the need to expand the curriculum with dedicated HPC courses and practical training opportunities, ensuring that graduates are well-prepared to meet the evolving demands of computational science, AI, and interdisciplinary research.

REFERENCES

[1] Weronika Filinger, Julie Mullen, Jeremy Cohen, Samantha Wittke, and Ann Backhaus. 2024. Building HPC Learning Pathways: Understanding our Community. In *Practice and Experience in Advanced Research Computing 2024: Human Powered Computing (PEARC '24)*. Association for Computing Machinery. https://doi.org/10.1145/3626203.3670513

[2] Michael Gowanlock. 2025. Teaching parallel and distributed computing using data-intensive computing modules. *Journal of Parallel and Distributed Computingournal of Parallel and Distributed Computing* 202 (2025), 105093. https://doi.org/10.1016/j.jpdc.2025.105093

[3] Bryan Johnston, Lara Timm, David Macleod, John Poole, Lily de Melo, Sayfullah Jumoorty, Jonathan Faller, Reinhard Jansen van Vuuren, and Michael Beukman. 2025. From Student SIG to Success: The journey of a student HPC Special Interest Group towards sustainable training and success in Student Cluster Competitions. *The Journal of Computational Science Education* 16 (03 2025), 43–49. https://doi.org/10.22369/issn.2153-4136/16/1/9

[4] Sushil K. Prasad, Almadena Chtchelkanova, Sajal Das, Frank Dehne, Mohamed Gouda, Anshul Gupta, Joseph Jaja, Krishna Kant, Anita La Salle, Richard LeBlanc, Manish Lumsdaine, David Padua, Manish Parashar, Viktor Prasanna, Yves Robert, Arnold Rosenberg, Sartaj Sahni, Behrooz Shirazi, Alan Sussman, Charles Weems, and Jie Wu. 2011. NSF/IEEE-TCPP curriculum initiative on parallel and distributed computing: core topics for undergraduates. In *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education*. Association for Computing Machinery, 617–618. https://doi.org/10.1145/1953163.1953336

Experience and Outcomes Organizing a Hackathon in the Physical Sciences

Aaron Jezghani†
Georgia Institute of Technology
ajezghani3@gatech.edu

Jason Fry†
Eastern Kentucky University
jason.fry@eku.edu

ABSTRACT

Despite its growing importance in physical sciences, research computing with cluster resources remains difficult to access and sustain, especially in long-term, multi-institutional projects. Challenges include site-specific workflows, evolving software stacks, and rapid changes in hardware post-Generative AI. The Nab collaboration, conducting a precision test of the Standard Model at Oak Ridge National Laboratory, hosted a hackathon to address these issues. Over four half-days, 25 participants engaged in training and collaborative problem-solving across four priority areas, supported by mentors and structured sessions. Post-event surveys showed improved computational knowledge and strong interest in recurring events. This paper shares insights from organizing the hackathon and discusses scalable strategies for computational training in experimental research.

KEYWORDS

Data Science, Hackathon, Cluster Computing

1 INTRODUCTION

The physical sciences have historically garnered attention for their impressive experimental setups and apparatuses: from tools such as macromolecular crystallography used in the development of new drugs [9] to the stunning imagery captured by the James Webb Science Telescope, scientists and engineers have demonstrated incredible ingenuity in developing tools to explore all scales of the universe around us. In spite of the primary focus on the design and outcomes from the experiment itself, most efforts today include a heavy component of computational effort, whether through the use of computer simulations to understand the operation of the experiment, or in the analysis of the data taken along the way. As an example of the ubiquitousness of research computing as a critical component of science and engineering, Figure 1 shows the top 25 domains served by the NSF XSEDE and ACCESS programs from July 2021 through July 2025, with nearly all cycles going towards science and engineering [2]; specifically, the physical science domains consume roughly 75% of all cycles.

† On behalf of the Nab collaboration.

https://doi.org/10.22369/issn.2153-4136/17/1/9

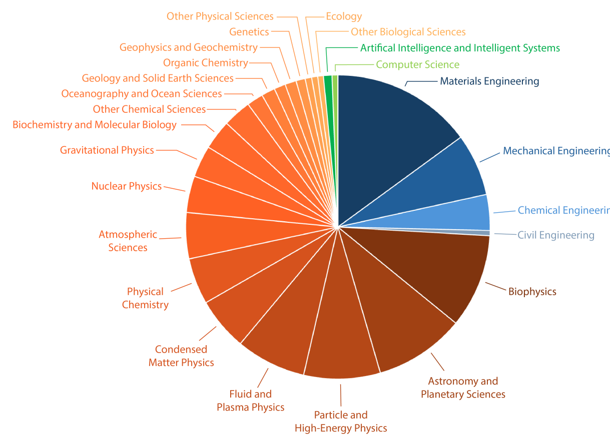


Figure 1: Top 25 domains by CPU hours across XSEDE/ACCESS systems from July 2021 through July 2025.

Note: Although engineering (blue) represents the greatest consumption of resources by a single domain, the physical sciences (orange) represent roughly 75% of all consumed cycles. Nuclear physics ranks 10th *in total consumption.*

In spite of the large volume of resources consumed by domain science users, they are not always the most efficient users in their work. Multiple reasons can explain this apparent contradiction, but one of the most likely culprits lies in the fact that as experimentalists in the domain sciences, the experimental apparatuses and the mastery of the theories in their specific domain are prioritized over the computational tools leveraged for research. For example, despite the domain of nuclear physics ranking 10th overall in CPU cycle consumption, a sample program from the National Nuclear Physics Summer School shows that only 3 of 80 hours in the program explore computational methods [11]; while service providers know the invaluable contribution of big computational resources for science and engineering, the systems are a tool to produce output for many users. Consequently, additional approaches are necessary to engage and onboard users in these domains.

University and national HPC centers offer training across a spectrum of topics, ranging from basic cluster orientation to more advanced topics like distributed and GPU programming. For example, Pittsburgh Supercomputing Center has tallied more than 24,000 attendees for its workshops on parallel, distributed, and big data computing, and feedback has been overwhelmingly positive [16]. However, the same paper acknowledges that the resources are presented in a simplified fashion, and the examples are presented for a general audience, so the specific translation to another computing resource or for a specific field of science may still present a barrier for the end user.

Another opportunity for researchers is to participate in Open-Hackathons events, which are sponsored and supported by OpenACC members spanning industry, academia, and national labs [14]. Since 2014, the organization has hosted bootcamps and hackathons that pair researchers with expert mentors to optimize and scale their specific applications. While the outcomes from these events is exceptional, the commitment of mentors and resources, as well as the time required, limit the scalability of these hackathons. As such, the events have a limit on the number of teams, and the application process is competitive, which can present as a barrier to researchers still developing their workflows.

In summary, there exists a gap for many users across the physical sciences, which is problematic for the efficient use of research computing platforms, as they represent the largest consumer of resources. In this paper, we present an internally organized and hosted hacakthon to address this challenge head on, and summarize the outcomes and feedback so that it can serve as a model for others interested in taking a similar approach.

2 THE NAB EXPERIMENT

Proposed in 2007, the Nab experiment is currently taking production data at Oak Ridge National Laboratory is a precision test of the Standard Model of Particle Physics based on the decay of free neutrons [3, 7] . A beam of unpolarized neutrons enters a 7 m magnetic spectrometer, which measures the phase space of neutron β-decay through direct measurement of the resultant electron and proton in pixelated silicon detectors above and below the decay region. The goal for the experiment is to provide the most precise determination of two parameters in the equation describing the neutron decay rate, which can help resolve contention between physics models and observed results, as well as provide a sensitive probe for new physics.

Unlike complementary efforts such as high-energy experiments conducted using the Large Hadron Collider at CERN, where the primary challenge is generation of the greatest quantity of relativistic particle collisions that yield statistically significant exotic physics signals above background levels, high-precision experiments such as Nab emphasize a rigorous understanding of the various sources of experimental error to appropriately bound. For the purposes of Nab, the demand for computing is two-fold:

- simulations of billions of decay events, which currently require several hundred thousand CPU-hrs each, must be run many times to quantify the experimental uncertainties, and
- rigorous analysis of multiple petabytes of simulation and experimental data, which must be replayed repeatedly as analysis algorithms are continuously refined.

2.1 The Collaboration

As shown in Figure 2, the collaboration today includes 51 active members from 16 institutions across 4 countries. To date, 18 PhD students have published dissertations as part of the Nab effort, with another 16 currently conducting research. Additionally, a number of high school and undergraduate students have contributed to the collaboration, as well as several post-doctoral researchers, academic faculty, and staff scientists. The effort of the collaboration leverages a hybrid model, with the main experimental work conducted at the lab and simulation, analysis, remote shift work, and small-scale studies occurring at collaborator home institutions.

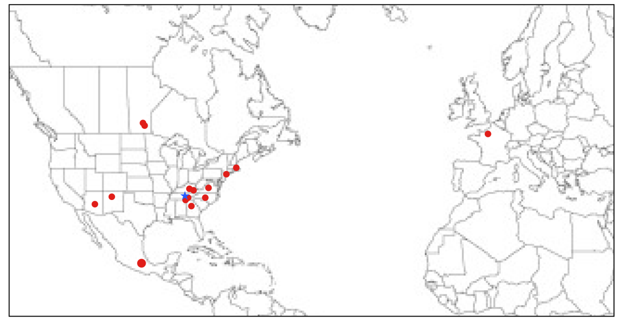


Figure 2: The Nab collaboration.

Note: The Nab collaboration is presently comprised of researchers from 16 institutions across 4 countries (red dots), and the experiment takes place at Oak Ridge National Laboratory (blue star).

Computing resources for Nab are distributed across various institutions. In addition to near-line analysis servers with GPUs and the physics division cluster, collaborating universities provide storage and computational cycles to support simulation and analysis efforts. In particular, computing effort on the Phoenix cluster at Georgia Tech (GT) is enabled by the institute-supported tier provided to all faculty as well as an Institute for Data Science and Engineering (IDEaS) allocation for CPU and GPU cycles plus 250TB of local storage for batch processing of the experimental data [8, 10]; cluster usage at GT since 2024 has totaled 1.6M CPU-hours and 6.2k GPU-hours.

2.2 Software Stack

Like many nuclear and particle physics experiments, the Nab software stack was built around CERN's analysis and simulation packages, `ROOT` and `Geant4`, respectively [1, 5]. With decades of software engineering baked into both applications, `ROOT` and `Geant4` both seek to address the challenges of efficiency, providing numerous template classes and interfaces to libraries for accelerated algorithms, efficient data structures, and routines for parallel and distributed processing. However, these capabilities present a double-edged sword, as building with the necessary dependencies is a non-trivial effort. In light of these challenges, Docker-based images and Spack installation instructions are provided; by default, however, these solutions are provided as separate containers, and enabling platform-specific plugins can still prove challenging. Additionally, the initial installation of `Geant4` provides only the libraries necessary to compile the Nab simulation, which itself introduces additional dependencies and complexities.

The Nab collaboration has developed an additional suite of tools for simulation and analysis for use in conjunction with the aforementioned packages:

- `Delta-Rice`, an HDF5 plug-in for compression of digitized waveform data, allows for extremely high-throughput compression and decompression [13],

- nabPy, a Python-based analysis stack with routines for waveform processing and physics analysis [12], and
- NESSE, a Python utility for detector response simulation [15].

Given the challenges to prepare the full swath of simulation and analysis software, an Apptainer container was developed to provide a portable, reproducible environment with the requisite software for simulation and analysis efforts for Nab. Starting from a base Ubuntu 20.04 image, Geant4 with support for ROOT data structures, as well as a compiled version of the Nab Simulation, was built and made available to collaborators. However, the use of the container was not well-documented, and the additional suite of Nab-specific packages was not included in the build, rendering it limited for use beyond a basic particle simulation. See Section 4.1 for improvements of the container made during the hackathon.

2.3 Motivations for Targeted Training

Although not formally tracked, internal communications have long indicated challenges in using cluster resources and software tools by various collaborators. Despite explicit requests for GT cluster access by 46 individuals, Figure 3 shows that only 21 users have accumulated any time since January 1, 2024. Furthermore, 85% of allocated CPU-hrs can be traced to just 2 of those users, while 70% of allocated GPU-hrs are the work of a single individual. Monthly utilization summaries collectively show utilization efficiency across the majority of users is less than 10% of theoretical limits.

As will be described in Section 3.1, participants were provided an opportunity for feedback following the collaboration hackathon. In particular, one question asked respondents to rank their skills in various topics before participation in the event. Figure 4 shows the results from the 15 individuals who provided feedback. Notably, skills such as Python scripting or cluster access via remote command-line interface or browser gateways ranked highest, while confidence in the Nab software stack, and particularly emergent GPU and AI workflows, were demonstrably lower.

In response to the increasing urgency for more effort and results from both simulation and analysis, the collaboration was encouraged by sponsoring agencies to address gaps in capability as expeditiously as possible. As such, Nab leadership decided to organize a dedicated event to facilitate knowledge sharing, especially to newer members, as well as expend effort on outstanding tasks.

3 THE NAB HACKATHON: “NABATHON”

Following the efficacy of the OpenHackathons format and the successes of participating teams, the Nab hackathon was fashioned similarly with a few modifications. First, the event was designed to be inclusive of all skill-levels; beyond development and optimization of specific applications, the event was meant to introduce new collaborators to the available tools and reinforce software development and maintenance practices across all . Second, the length of the event, both in number of days as well as the length of each day, had to be reduced due to other scheduled activities. Lastly, the results were inclusive of physics outcomes as well as improvements to specific software packages and computational workflows.

3.1 Organization

Organization of the hackathon began approximately six weeks in advance of the event. Based on the experimental schedule, data-taking campaigns, and outstanding tasks, senior personnel from the collaboration identified simulation and analysis priorities to assign to subteams for the event. For each task, one or more mentors with expertise in the physics or software tools were identified to shepherd the subteam effort; additionally, one mentor with general cluster knowledge provided guidance across all subteams as relevant. This list of tasks was then circulated to the collaboration as a Google spreadsheet, asking members to identify subteams to which they were conducting synergistic activity and could contribute.

Prior to the start of the hackathon, portions of weekly collaboration analysis calls were dedicated to discussion of specific details of the hackathon tasklist. Logistics for the hybrid event, participant expectations, and cluster access were also addressed during this period. Zoom meeting invitations were prepared for remote attendees, while co-located attendees were able to reserve a lab conference room for the event. Additionally, sample simulation macros and datasets were prepared to provide a baseline for effort during the hackathon.

The event was scheduled as four half-days over a period of one week, with an introductory Day 0 on Thursday for cluster onboarding and final subteam selection, and three hacking days starting the following Monday. Participants were encouraged to engage asynchronously before the first hacking day to familiarize themselves with the cluster resources and subteam tasks. In addition to existing Phoenix cluster access, participants were given temporary accounts on the instructional cluster [4, 6], which provided the same software stack and hardware architectures, but with more conservative resource limits to reduce queue wait times for teaching purposes.

Day 0 was scheduled from 9 am - 1 pm and content was crowd-sourced from existing GT training, course content, and publicly-available community content; hosted on the Nab collaboration internal git repository with links for recording of day 0 content. The git repository also housed directories for training and homework for participants. The scrum and final presentations were accessed through a in shared google drive with templates for each that included skeleton structure for content and recommended presentation time.

At the start of Days 1 and 2, teams presented progress, goals, and obstacles as part of morning scrums. Afterward, the teams hacked from roughly 9:20 am - 1 pm. At the end of Day 3, teams gave a longer presentation reporting on their final results and status of assigned activity. Additionally, an anonymous survey was disseminated via Microsoft Forms to participants to provide feedback on the efficacy of the event, including both the organization and structure as well as the technical components.

3.2 Subteam Efforts

In pre-planning for the hackathon during collaboration analysis calls, three main efforts were chosen and subteams were assigned. The three subteams were simulation and pipeline, a-fitting, and magnetometry. A spreadsheet was sent around for everyone participating to sign up for the various groups and subgroups and expert mentors were identified for each subgroup. The simulation and

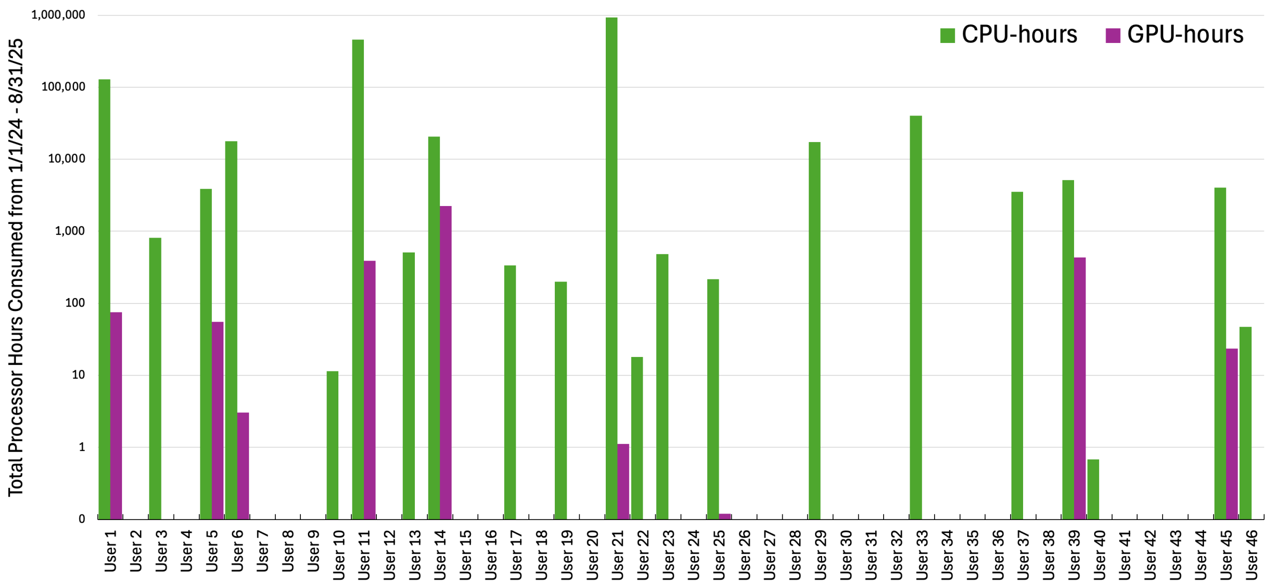


Figure 3: CPU-hrs and GPU-hrs consumed by Nab collaborators on the Phoenix cluster.

Note: Of the 46 accounts that were requested, only 21 have any trackable activity on the cluster, with most effort dominated by just a few users.

pipeline team focused on the geant4 and NESSE simulations with the goal of creating a pipeline between the two. The magnetometry group focused on the analysis on the mapping of the complex magnetic fields in the Nab spectrometer. This included traditional analysis in python and C++ as well as a neural network approach. The a-fitting group's focus was on developing new algorithms to fit the complex 2D phase space that Nab measures as well as further developing existing algorithms.

4 OUTCOMES AND FEEDBACK

As described in section 3.1, a survey was created and sent to all participants of the hackathon with 15 respondents to assess pre-event and post-event proficiency. Overall, the survey showed that the participants found the event useful and the collaboration agreed hackathons would be useful going forward on as annual or semiannual events.

4.1 Technical Accomplishments

Figures 4 and 5 show the participants proficiency in various computing areas before and after the hackathon. We can see that the mean reported values of the all the areas increased and the Nab-based areas increased significantly. Improvements were readily apparent in the Nab software stack, the use of software containers, and the development of more effective workflows with Slurm. However, little improvement was demonstrated for GPU and AI activity, which can be attributed to a lack of organized effort and focus on those two particular topics.

While the survey results indicated pre-event and post-event proficiency, the subteams presented physics achievements in the final slide decks. In the simulation and pipeline group, the team was able to marry two previously separate software packages to

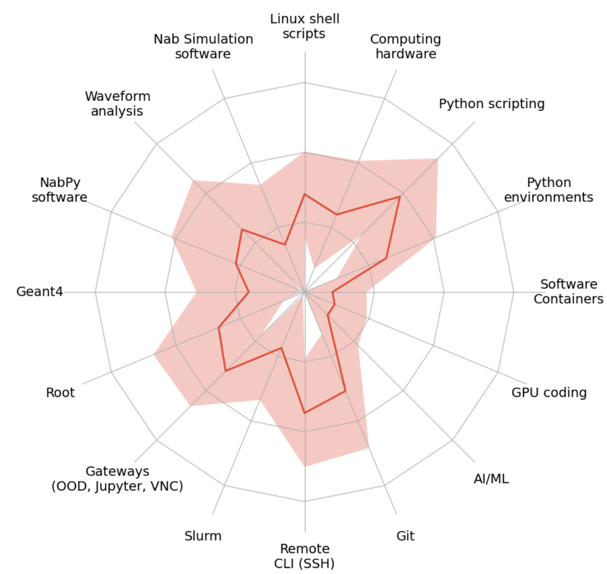


Figure 4: Self-reported proficiency by participants prior to the hackathon event.

Note: Radial distance represents participant expertise in a particular topic, with the line and band representing the mean reported value plus or minus one standard deviation, respectively. Despite moderately high confidence in fundamental skills such as Python scripting, version control, and basic command line interface, respondents did not express strong confidence in neither the Nab software stack nor novel workflows using GPUs or AI.

form a pipeline between the two. This was one of the main objectives of the hackathon and the team produced documentation of

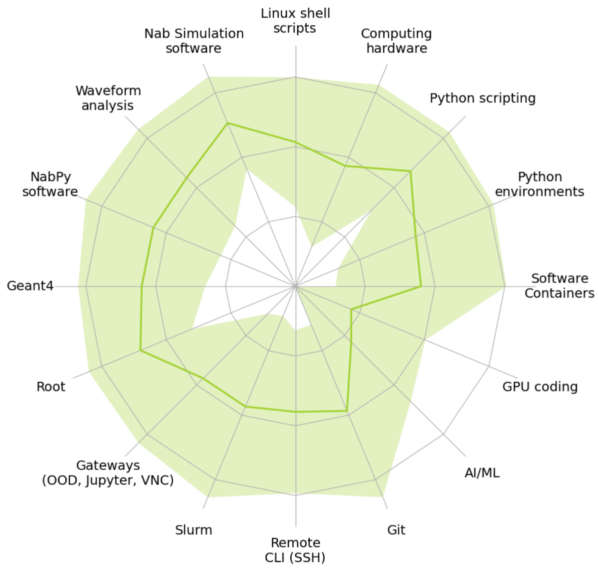


Figure 5: Self-reported proficiency by participants following the hackathon event.

Note: Increasing distance from the center represents increasing expertise. The line represents the mean for the data while the band shows one standard deviation about the mean. Respondents reported on average significant improvements in collaboration software, Slurm resource allocation, and software containers; however, the wide band reflects a broad range of confidence from participants.

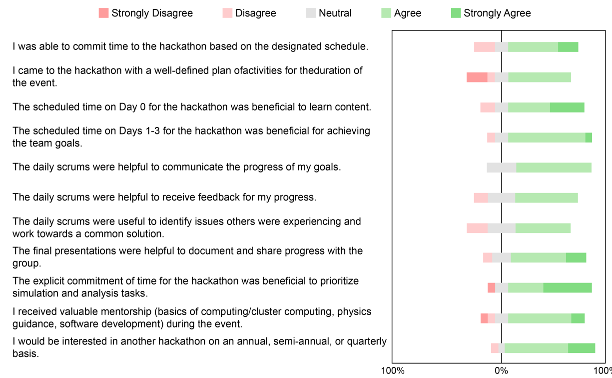


Figure 6: Participant survey results.

Note: According to survey respondents, the overall perception of the event was quite positive. Notable detractors include a lack of explicit effort, the efficacy of the daily scrums, and the quality of mentorship, but in theory these could be addressed with better planning. Despite this, respondents agreed that recurring events would be invaluable for the collaboration.

the changes implemented. The group also produced an 10^8 event simulation in 18 hours using 500 CPUs through slurm. Now the collaboration has a set of slurm templates for submitting large simulation and analysis jobs. In the magnetometry group, the team used the dedicated time to devote to developing analysis algorithms of the magnetic field data. This analysis produced a data-driven approach to making the next measurements of the magnetic field. The a-fitting group focused on fitting algorithms in python (such as `lmfit`) and compared speed and accuracy of different fitting infrastructure as well as documenting all the existing fitting algorithms of the collaboration.

4.2 Reflection on the Hackathon

The survey also included a question about what computational resources were utilized during the hackathon, which allowed multiple responses: personal device, GT instructional compute cluster, and GT Phoenix compute cluster. Out of the 15 participants, 5 of the participants worked exclusively on their personal devices for the event. Of the remaining 10 that took advantage of GT compute clusters, 5 used both Phoenix and the instructional cluster, 3 used only the instructional, and 2 used only Phoenix cluster.

In the open-ended question asking for any additional feedback, multiple respondents indicated the need for a better job organizing subteams and tasks, especially the explicit definition of deliverables at the end of the hackathon. Another respondent also suggested that the Day 0 content could be more impactful for participants if organized as a hands-on workshop rather than presented via slide decks. Yet another person suggested that increased in-person participation would be beneficial, although logistically, this is perhaps the most challenging component to address given the distribution of collaboration members. Nonetheless, 13 of the 15 respondents concurred that there should be another hackathon event within the next 6 to 12 months, with commentary that the event was particularly effective for advancing their research efforts within the collaboration.

5 CONCLUSION

Despite the prominence of users from the physical sciences on university and national research computing platforms, many experimentalists de-prioritize computational training and knowledge for attention to hands-on effort and domain expertise. Following a popular model for project-oriented hackathons, the Nab collaboration recently organized an event to streamline new student onboarding, expedite software development, and improve resource utilization efficiency. While multiple participants commented that the hackathon organization could be improved, with more detailed task identification and subteam effort, participant responses were largely positively to the event. Self-reported technical proficiency in cluster utilization, hardware, and software improved in many areas, while the general sentiment was favorable for recurring events to continue the successes following the 4 day event.

We feel that similarly self-organized events can provide a scalable and reproducible mechanism to enhance the computational abilities for other experimentalists who consume large quantities of computer resources. In addition to collaborations organizing such events to drive their own efforts, disjoint groups with common interests, particularly across multiple universities, can collectively arrange for their own hackathons to maximize the return on investment, without the barrier to entry they may encounter in other events. Additionally, we can imagine that this may reduce the overall support burden for cluster administrators, as researchers are exposed to the tools for the specific problems

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation (NSF) Contract PHY-2412846. This research was supported through research cyberinfrastructure resources and services provided by the Partnership for an Advanced Computing Environment (PACE) at the Georgia Institute of Technology, Atlanta, Georgia, USA.

REFERENCES

[1] 2003. Geant4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506, 3 (2003), 250–303. https://doi.org/10.1016/S0168-9002(03)01368-8

[2] ACCESS CI Database. 2025. ACCESS-CI XDMoD: CPU Hours: Total: by Field of Science: 2021-07-01 to 2025-07-31. https://xdmod.access-ci.org/#tg_usage?node=statistic&realm=Jobs&group_by=fieldofscience&statistic=total_cpu_hours

[3] Ricardo Alarcon, Septimiu Balascuta, Andreas Klein, W. Scott Wilburn, Michael T. Gericke, John R. Calarco, F. William Hersman, Albert Young, James D. Bowman, T. Vince Cianciolo, Seppo I. Penttila, Krzysztof P. Rykaczewski, Glenn R. Young, Vladimir Gudkov, Geoffrey L. Greene, Robert K. Grzywacz, L. Peter Alonzi, Stefan Baeßler, Maksim A. Bychkov, Emil Frlez, Anthony Palladino, and Dinko Pocanic. 2007. Precise Measurement of the Neutron Beta Decay Parameters "a" and "b". http://nab.phys.virginia.edu/nab_proposal.pdf

[4] Mehmet Belgin, Trever C. Nightingale, David A. Mercer, Fang Cherry Liu, Peter Wan, Andre C. McNeill, Ruben Lara, Paul Manno, and Neil Bright. 2018. ICE: A Federated Instructional Cluster Environment for Georgia Tech. In *Proceedings of the Practice and Experience on Advanced Research Computing: Seamless Creativity (PEARC '18)*. Association for Computing Machinery, New York, NY, USA, Article 16, 7 pages. https://doi.org/10.1145/3219104.3219112

[5] Rene Brun and Fons Rademakers. 1997. ROOT — An object oriented data analysis framework. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 389, 1 (1997), 81–86. https://doi.org/10.1016/S0168-9002(97)00048-X New Computing Techniques in Physics Research V.

[6] J. Eric Coulter, Michael D. Weiner, Aaron Jezghani, Matthew Guidry, Ruben Lara, Fang (Cherry) Liu, Allan Metts, Ronald Rahaman, Kenneth Suda, Peter Wan, Gregory Willcox, Deirdre Womack, and Dan (Ann) Zhou. 2023. ICE 2.0: Restructuring and Growing an Instructional HPC Cluster. In *Proceedings of the SC '23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis (SC-W '23)*. Association for Computing Machinery, New York, NY, USA, 591–597. https://doi.org/10.1145/3624062.3624131

[7] Jason Fry, Ricardo Alarcon, Stefan Baeßler, Septimiu Balascuta, Libertad Barrón Palos, Thomas Bailey, Kevin Bass, Noah Birge, Alexander Blose, Dennis Borissenko, James D. Bowman, Leah J. Broussard, Albert T. Bryant, James Byrne, John R. Calarco, Jimmy Caylor, K. Chang, Timothy Chupp, T. Vince Cianciolo, Christopher Crawford, Xinjian Ding, Margaret Doyle, Wenjiang Fan, W. Farrar, Nadia Fomin, Emil Frlež, Michael T. Gericke, Michelle Gervais, Ferenc Glück, Geoffrey L. Greene, Robert K. Grzywacz, Vladimir Gudkov, Joshua Hamblen, Christopher Hayes, Chelsea Hendrus, Takeyasu Ito, Aaron Jezghani, Huangxing Li, Mark Makela, Nicholas Macsai, Juliette Mammei, Russell Mammei, Madelyn Martinez, David G. Matthews, Mark McCrea, Patrick McGaughey, C. David McLaughlin, Paul Mueller, D. van Petten, Seppo I. Penttilä, David E. Perryman, Rüdiger Picker, Josh Pierce, Dinko Počanić, Yu Qian, John Ramsey, Glenn Randall, Grant Riley, Krzysztof P. Rykaczewski, Americo Salas-Bacci, Sepehr Samiei, E. Mae Scott, Thomas Shelton, Skye K. Sjue, Alexander Smith, Erick Smith, Eric Stevens, Jonathan Wexler, Ryan Whitehead, W. Scott Wilburn, Albert Young, and Bryan Zeck. 2019. The Nab experiment: A precision measurement of unpolarized neutron beta decay. *EPJ Web Conf.* 219 (2019), 04002. https://doi.org/10.1051/epjconf/201921904002

[8] IDEaS. 2025. Institute for Data Engineering and Science. https://research.gatech.edu/data

[9] Mariusz Jaskolski, Zbigniew Dauter, and Alexander Wlodawer. 2014. A brief history of macromolecular crystallography, illustrated by a family tree and its Nobel fruits. *The FEBS Journal* 281, 18 (2014), 3985–4009. https://doi.org/10.1111/febs.12796 arXiv:https://febs.onlinelibrary.wiley.com/doi/pdf/10.1111/febs.12796

[10] Aaron Jezghani, Semir Sarajlic, Michael Brandon, Neil Bright, Mehmet Belgin, Gergory Beyer, Christopher Blanton, Pam Buffington, J. Eric Coulter, Ruben Lara, Lew Lefton, David Leonard, Fang Cherry Liu, Kevin Manalo, Paul Manno, Craig Moseley, Trever Nightingale, N. Bray Bonner, Ronald Rahaman, Christopher Stone, Kenneth J. Suda, Peter Wan, Michael D. Weiner, Deirdre Womack, Nuyun Zhang, and Dan Zhou. 2022. Phoenix: The Revival of Research Computing and the Launch of the New Cost Model at Georgia Tech. In *Practice and Experience in Advanced Research Computing 2022: Revolutionary: Computing, Connections, You (PEARC '22)*. Association for Computing Machinery, New York, NY, USA, Article 13, 9 pages. https://doi.org/10.1145/3491418.3530767

[11] Jinfeng Liao, Walter Pettus, W. Michael Snow, and Scott Wissink. 2024. National Nuclear Physics Summer School. https://ceem.indiana.edu/events/nnpss/program.html

[12] David G. Mathews. 2025. pyNab Repo. https://gitlab.com/NabExperiment/pyNab

[13] David G. Mathews, Christopher B. Crawford, Stefan F. Baessler, Noah Birge, Leah J. Broussard, Francisco M. Gonzalez, Leendert Hayen, Aaron Jezghani, Huangxing Li, Russell R. Mammei, August Mendelsohn, Glenn Randall, Grant V. Riley, and Danielle C. Schaper. 2024. Delta-Rice: A HDF5 Compression Plugin optimized for Digitized Detector Data. *Journal of Open Source Software* 9, 98 (2024), 6598. https://doi.org/10.21105/joss.06598

[14] OpenHackathons. 2025. OpenHackathons Webpage. https://www.openhackathons.org/s/

[15] Robert J. Taylor. 2025. Nab Event Shape Simulation Effort (NESSE) Repo. https://github.com/rjtayl/nesse

[16] John Urbanic, Thomas Maiden, and Valerie Rossi. 2024. Evaluating the Wide Area Classroom After 24,000 HPC Students. arXiv:physics.ed-ph/2402.03343 https://arxiv.org/abs/2402.03343

Building Scalable and Inclusive Foundations for HPC: Lessons from UC Merced's Introductory HPC Training Program

Yue Yu
University of California, Merced
yyu49@ucmerced.edu

ABSTRACT

High-performance computing (HPC) is becoming essential across a broad range of disciplines, including those historically underrepresented in computational research, such as sociology, psychology, and the arts. To reduce barriers to entry, the University of California, Merced (UC Merced) developed a 90-minute introductory HPC workshop designed for participants with no prior technical background. The workshop includes a theoretical overview of campus clusters, fundamental Linux commands, and core HPC concepts, followed by a hands-on session where participants connect through SSH and browser-based tools, load software modules, and submit jobs to institutional HPC resources using Slurm. Delivered in a hybrid format with both synchronous and asynchronous learning materials, the program has been offered more than 20 sessions since 2021. It has primarily served students (75.7%), faculty (16.2%), and staff. Post-workshop surveys indicate that 83% of participants are more likely to incorporate HPC into their research after attending, contributing to a doubling of active HPC users on campus since the program's launch. This scalable and inclusive model provides an effective framework for expanding HPC adoption and fostering computational engagement across disciplines.

KEYWORDS

HPC Training, Workshop Design, Hybrid Learning, Student Engagement

1 INTRODUCTION

High-performance computing (HPC) has become a critical component across a wide range of disciplines. It allows researchers to simulate complex systems where traditional experiments are infeasible, analyze massive datasets that exceed the capabilities of personal computers, and develop advanced computational tools to improve predictions and decision-making. In recent years, there has been rapid growing interest in extending HPC to fields historically underrepresented in computational research, such as social sciences, humanities, and psychology. However, adoption in these fields remains limited for several reasons. First, there is a lack of foundational training materials designed for researchers without technical backgrounds, which can make HPC seem intimidating or out of reach. Second, awareness of available HPC resources is often low, especially in disciplines that historically have not used computational approaches. Third, there are cultural and methodological barriers that researchers may not see computational methods as aligned with their field, or they have the perception that HPC is overly complex or reserved for specialized fields.

To address these challenges, the University of California, Merced (UC Merced) developed a 90-minute introductory HPC workshop aimed at lowering barriers and making computational resources more approachable for researchers from diverse backgrounds. Based on the principles of inclusive pedagogy and experiential learning, the workshop blends conceptual instruction with hands-on practice. A key enhancement is a virtual tour of the campus data center, which allows participants to visualize the physical infrastructure behind the HPC system, fostering familiarity and reducing the intimidation often associated with large-scale computing facilities. This approach ensures that attendees not only gain a foundational understanding of HPC concepts but also build confidence and practical skills that they can immediately apply to their own research.

This paper presents the workshop as a case study in effective HPC onboarding, detailing its structure, hybrid delivery format, and support strategies. We evaluate its impact on broadening HPC participation among historically underrepresented disciplines and regions. By placing this initiative within the wider landscape of HPC education, we aim to provide a replicable model that other similar institutions can adopt to democratize access to advanced computational resources and foster a more diverse research community.

2 WORKSHOP DESIGN

The UC Merced introductory HPC training program was designed to lower barriers for new users, particularly those from non-technical or historically underrepresented disciplines. The program is grounded in educational best practices for teaching complex technical concepts to diverse audiences, emphasizing active participation, inclusivity, and confidence-building.

2.1 Educational Principles

The workshop includes several core educational principles to ensure both effectiveness and accessibility.

- Active Learning: Participants engage directly with HPC concepts through hands-on practices where they will be able to use UC Merced HPC to complete the tasks.
- Peer Support: Trained student assistants provide real-time guidance, fostering a collaborative and supportive learning from their peers.
- Scaffolded Learning: Concepts are introduced progressively, allowing participants to build knowledge step-by-step without becoming overwhelmed.

https://doi.org/10.22369/issn.2153-4136/17/1/10

- Hybrid Teaching Model: Combines in-person sessions with asynchronous learning materials, ensuring participants have ongoing access to resources. This approach reduces barriers to entry and allows learners to review content at their own pace, supporting deeper understanding and retention of new concepts.

2.2 Workshop Structure

The workshop begins with a theoretical introduction to HPC, starting with an overview of what HPC is and how it differs from other scales of computing, such as personal or lab-scale desktops. Participants are introduced to fundamental terminology, including definitions of cores, CPUs, and nodes, to help them become familiar with the language and concepts that will be used throughout the session. Once these key terms are understood, the discussion moves to hierarchical computing structures, illustrating how these components fit together in modern HPC systems.

Next, participants are introduced to UC Merced's on-premises HPC clusters, including their total number of nodes and cores, to demonstrate the scale of computational resources available on campus. By comparing this to personal computing capabilities, attendees gain motivation to leverage HPC for accelerating their research. The session then covers the general architecture of an HPC system, explaining key components such as login nodes, compute nodes, SSH access, and high-speed interconnects like InfiniBand. This provides participants with a clear picture of how different parts of the cluster work together to execute programs.

To connect theory with practice, the workshop includes a virtual tour of the campus data center, giving participants a view of the physical layout of servers and supporting infrastructure. This visual component helps demystify HPC by illustrating how abstract concepts translate into real-world hardware, fostering confidence that these systems are both approachable and accessible. This concludes the theoretical portion of the workshop, highlighted in purple in Figure 1.

After the theoretical section, the workshop moves into a hands-on component (shown in green in Figure 1), where participants actively interact with the HPC system. To reduce cognitive load, each participant receives a Linux command cheat sheet containing only the commands relevant to the session. This focused approach prevents users from being overwhelmed by extraneous details and helps them concentrate on applying commands in context. Throughout the exercises, participants can refer to the cheat sheet as needed, allowing them to stay engaged without worrying about memorization or losing track during the session.

Participants use guest accounts created specifically for the workshop. These accounts are configured with restricted permissions and access only to designated public queues, ensuring that training activities do not disrupt overall cluster operations. Each guest account includes a preconfigured home directory with folders and files prepared for the practice session, streamlining the experience and removing setup barriers. Temporary passwords are provided so participants can log in via SSH, offering an authentic introduction to remote HPC access.

In addition to SSH, participants are introduced to Open OnDemand [4], a browser-based interface for interacting with the cluster. This tool allows them to launch a terminal directly from their web browser using their guest credentials, further lowering technical barriers by removing the need to install or configure terminal software locally. This dual-access approach supports participants with different technical skills and resources, ensuring an inclusive experience.

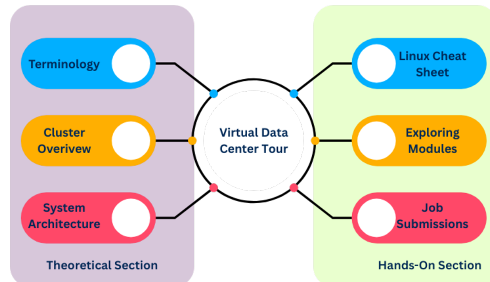


Figure 1: The general workflow of the workshop.

Once connected, participants start by exploring the basic cluster environment. They are introduced to commands such as `'module avail'` and `'module load'`, which allow them to view and access software modules already available on the cluster. This step encourages participants to develop the habit of checking for pre-installed resources before building or installing their own, promoting both efficiency and reproducibility in their workflows.

After exploring the cluster environment, the session transitions to running programs and submitting Slurm [9] jobs onto institutional HPC cluster, introducing participants to the concept of job scheduling and resource management. They begin with serial job exercises using simple, pre-configured scripts. The process starts by examining the contents of a basic Slurm job script with the `'cat'` command. This method allows participants to view the script directly in the terminal without the added complexity of using a text editor. Special attention is given to lines starting with `'#SBATCH'`, which define job parameters such as runtime, memory allocation, and partition selection. This helps participants understand how Slurm works and how to properly request computational resources. After the script review is complete, participants submit their first job using the `'sbatch'` command and monitor its progress with `'squeue'`. Observing job behavior and output file generation in real time reinforces the connection between Slurm commands and the underlying cluster operations. This hands-on practice builds confidence as participants see their jobs successfully run on the system.

Building on this foundation, the session advances to parallel job exercises, introducing more complex workflows. Here, participants work with scripts that submit multiple serial jobs within a single Slurm script. They are introduced to loop syntax in Bash, which allows them to iterate over tasks and automate multiple job submissions seamlessly. These exercises include node count, core allocation, partition settings, runtime limits, and environment variables. By working through these examples, participants begin to understand how to scale their computations and optimize resource use.

This structured progression begins with basic Linux navigation, continues with single-job submissions, and culminates in automated multi-job workflows, offering participants a practical and accessible introduction to HPC job management. By the end of the session, they had gained both the knowledge and the confidence to independently perform real-world research computations on the cluster.

2.3 Delivery Format

The workshop is designed as a 90-minute session, divided evenly between two parts: the first 45 minutes focus on the theoretical introduction, followed by 45 minutes of hands-on practice. This duration was chosen to align with both the workshop content and the average human attention span reported in educational research [1, 7]. Keeping the session under two hours helps prevent participant fatigue and avoids overwhelming new users while still providing sufficient time for meaningful learning.

To maximize accessibility and support long-term learning, the workshop is offered in both synchronous and asynchronous formats. Live sessions give participants the opportunity to interact with instructors and student assistants, ask questions, and receive real-time guidance. Afterward, all materials are made available through an open-access HPC documentation website [8].

The documentation site has been in continuous development since 2022. It was originally built with Docsify [3] but later migrated to Docusaurus [5], a modern framework developed by Meta. Docusaurus was chosen for its native support of React components, offering greater customization, flexibility, and scalability for creating interactive documentation. This transition also follows best practices recommended by other universities [2], ensuring the platform remains sustainable and user-friendly for a broad research audience.

The updated site now includes a dedicated training materials section (Figure 2) and a blog page for community updates, announcements, and job opportunities, helping participants stay engaged beyond the workshop. It also provides essential reference materials such as cluster configurations, command explanations, and troubleshooting guides, allowing users to revisit content at their own pace. Training slides, example code, and step-by-step instructions are included to guide learners through transferring exercise files, running practice jobs, and reinforcing key concepts. This structure enables participants to continue developing their HPC skills independently long after the session has concluded.

2.4 Student Assistant Involvement

Currently, an undergraduate student serves as a Student Technology Consultant (STC) in the Office of Information Technology (OIT) at UC Merced. This student has contributed significantly to the migration and redevelopment of the HPC documentation website and material development, gaining valuable experience in web infrastructure and user support. Beyond documentation, they have been trained to lead hands-on practice sessions during workshops, participate in weekly HPC office hours, and assist with troubleshooting user issues through the ServiceNow ticketing system.

Through these experiences, the student has advanced from a novice HPC user to an independent and highly skilled practitioner. This growth is demonstrated by several notable accomplishments, including leading UC Merced's first student team in the IndyCC competitions at SC24 and SC25, as well as completing two summer internships with HPC and computational science groups at Los Alamos National Laboratory, further deepening their expertise.

Inspired by this success, we plan to expand student involvement by recruiting and training additional undergraduate assistants. To foster a sustainable, peer-supported learning environment, we recently launched a Slack platform dedicated to the campus HPC community. This space is student-led and designed to encourage open discussion, promote training opportunities, and facilitate resource sharing. Our goal is to build a collaborative community where students not only receive support but also take on leadership roles in advancing HPC engagement and knowledge-sharing across campus.

3 IMPLEMENTATION AT UC MERCED

3.1 Program History

The Introductory HPC Workshop was first launched in late 2021 as part of UC Merced's broader strategy to expand access to research computing resources and support the university's transition to R1 status. As UC Merced grows its research portfolio, empowering faculty, students, and staff with advanced computational skills is essential to fostering a more active and data-driven research community.

Since its launch, the program has steadily expanded, offering more than 20 sessions to participants across diverse disciplines. By delivering accessible, foundational HPC training, the workshop has enabled researchers to integrate advanced computational methods into their work, directly contributing to the growth of UC Merced's research portfolio. These initiatives have significantly enhanced the university's research capacity and were instrumental in supporting its achievement of R1 status in February 2025. Looking ahead, the program will continue to promote computational literacy and foster collaboration, ensuring that UC Merced not only maintains its R1 designation but also strengthens its role as a hub for cutting-edge research in California's Central Valley.

3.2 Workshop Schedule and Requests

We offer the HPC workshop on two fixed dates each year. The first session takes place during UC Merced Research Week, typically held in March, when the campus highlights active research initiatives. As part of this event, we host an in-person data center tour alongside the workshop to showcase the HPC facilities and demonstrate their role in supporting cutting-edge research. The second fixed session is held during Graduate Orientation Week (GROW), usually in August, when the campus welcomes new graduate students. Hosting the workshop at this time introduces incoming researchers to research computing early in their programs, encouraging them to integrate HPC resources into their work from the start.

In addition to these fixed sessions, we also conduct on-demand workshops. Requests can be submitted through the ServiceNow ticketing platform using the request form linked on the HPC documentation website. To maintain the quality of the hands-on learning experience, we require a minimum of three participants and limit each session to 25 attendees.

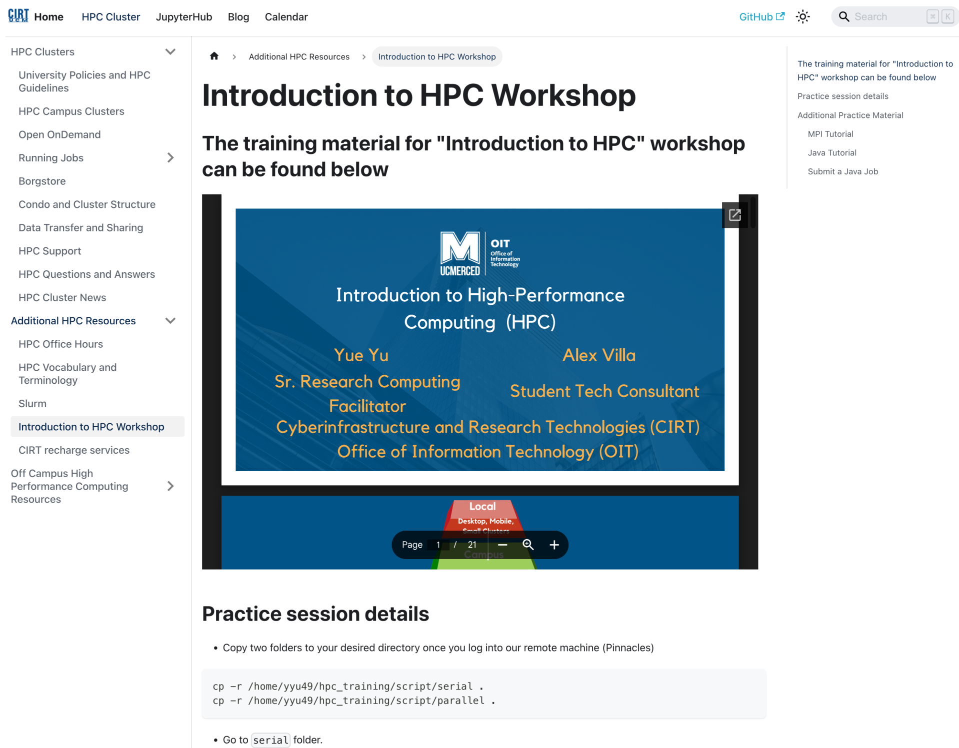


Figure 2: The HPC documentation website.

From these on-demand sessions, we have identified two common request patterns. The most frequent requests come from instructors who wish to integrate the HPC workshop into their courses at the beginning of each semester, particularly at the graduate level. This timing helps students become familiar with HPC early on so they can apply it to data analysis throughout the term. The second most frequent requests come from lab managers, who invite us to deliver the workshop during regular lab meetings. In these cases, the lab meeting is replaced with an HPC training session, providing a convenient and highly relevant learning experience for the entire research group.

3.3 Audience Profile

Across the 25 sessions delivered to date, the workshop has averaged 15 participants per session, with attendees consisting of 75.7% students, 16.2% faculty, and the remainder staff (see Figure 3). UC Merced, the most diverse campus in the University of California system, serves a student population in which the majority come from low-income households and are first-generation college students. The university is also designated as a Hispanic-Serving Institution (HSI), with Hispanic students representing more than 55% of the student body. This unique demographic underscores the importance of providing equitable access to advanced research computing resources, creating opportunities for groups historically underrepresented in computational research to engage with cutting-edge technologies.

Notably, 70% of workshop participants reported being completely new to HPC, emphasizing the program's role in onboarding first-time users. Post-workshop surveys further revealed that 83% of attendees felt more likely to incorporate HPC into their research, demonstrating the workshop's effectiveness in fostering computational engagement across a diverse campus community.

Participants particularly appreciated the hands-on format, which offered practical experience with real HPC systems. As one participant noted, "The course was amazing! I learned so much, not just about using the HPC, but also about working in the terminal and understanding the physical structure of the system. For a 1.5-hour presentation, I was very pleased with the content."

Since the program's launch, the number of active HPC users at UC Merced has doubled, reflecting its strong impact on campus-wide adoption. Collectively, these outcomes highlight the workshop's effectiveness as a scalable and inclusive model for HPC onboarding, advancing computational literacy and expanding research capacity across the university.

4 FUTURE DIRECTIONS

The long-term goal of this workshop program is to expand its reach beyond UC Merced, providing accessible and inclusive HPC training to a broader audience across the Central Valley of California.

4.1 Regional Scaling and Collaboration

We aim to scale the program to other institutions, including community colleges, California State University (CSU) campuses, and

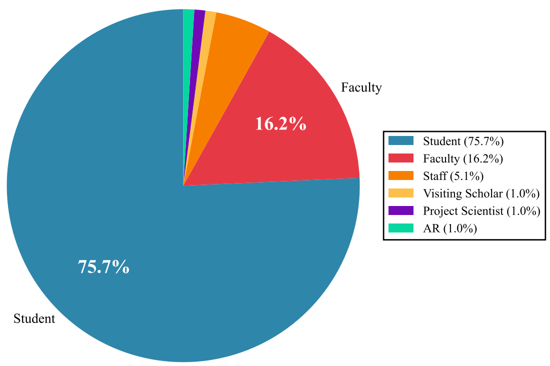


Figure 3: Affiliation of workshop attendees at UC Merced.

additional University of California campuses. The geographic proximity of UC Merced, CSU Sacramento, CSU Stanislaus, and CSU Fresno creates a unique opportunity for collaboration and shared resources. These institutions are not only physically close but also share a common mission as HSIs, which strengthens their collective commitment to advancing equitable access to research computing and fostering diversity in computational research.

- CSU Sacramento – Sacramento, approximately 120 miles north of UC Merced
- CSU Stanislaus – Turlock, approximately 40 miles northwest of UC Merced
- CSU Fresno – Fresno, approximately 85 miles south of UC Merced

By fostering partnerships across these campuses, we aim to establish a regional HPC training hub that supports the goals of the NSF award (#2346744) [6]. This hub will provide accessible solutions to empower historically underserved and geographically isolated regions with limited access to computational resources and collaboration opportunities. Through these efforts, we seek to overcome barriers, help researchers effectively leverage advanced computing tools, and foster inclusivity while expanding research capabilities. The hub will also serve underrepresented communities, promote interdisciplinary collaborations, and create pathways for students and researchers to engage with cutting-edge computational technologies and resources.

4.2 Developing a Modular HPC Curriculum

To meet the evolving needs of participants, we plan to expand the workshop curriculum into a modular, multi-level format, covering introductory, intermediate, and advanced HPC topics. The current workshop primarily serves as an entry point, introducing basic HPC concepts and workflows to first-time users. However, as the number of trained users grows, there has been an increasing demand for more advanced and domain-specific content.

Ongoing efforts include the development of intermediate training materials, focusing on:

- Best practices for parallel programming and workflow scaling.
- Job optimization strategies to improve efficiency and resource utilization.
- Hands-on projects tailored to specific research domains, such as bioinformatics, computational chemistry, and machine learning.

This modular approach enables participants to advance through progressively higher levels of HPC training, fostering essential skill development and building a pipeline of proficient users. By gaining the knowledge and experience to effectively utilize HPC resources, participants can enhance their research productivity and become more competitive in their future academic and professional careers.

5 CONCLUSION

UC Merced's introductory HPC workshop has proven effective in lowering barriers to computational research, with over 20 sessions delivered since 2021. The program has successfully engaged a diverse audience, 70% of whom were first-time HPC users, and has doubled the number of active HPC users on campus. Post-workshop surveys show that 83% of participants are more likely to integrate HPC into their research after attending.

Looking ahead, we plan to expand regionally, building a Central Valley HPC training hub in partnership with nearby institutions, and to develop a modular curriculum that supports sustained skill development from introductory to advanced levels. This initiative demonstrates how inclusive, scalable training can grow HPC adoption and empower researchers to leverage advanced computing for impactful, interdisciplinary research.

REFERENCES

[1] Scott Freeman, Sarah L. Eddy, Miles McDonough, Michelle K. Smith, Nnedi Okoroafor, Hannah Jordt, and Mary Pat Wenderoth. 2014. Active Learning Increases Student Performance in Science, Engineering, and Mathematics. *Proceedings of the National Academy of Sciences* 111, 23 (2014), 8410–8415. https://doi.org/10.1073/pnas.1319030111

[2] Ben Godfrey. 2024. Practical Advice for the Creation of Effective HPC User Documentation. In *Proceedings of PEARC '24: Practice and Experience in Advanced Research Computing*. Association for Computing Machinery, 83:1–83:4. https://doi.org/10.1145/3626203.3670621

[3] QingWei Hu. 2023. Docsify: A Magical Documentation Site Generator. https://docsify.js.org/. Accessed: 2025-09-05.

[4] Dave Hudak, Doug Johnson, Alan Chalker, Jeremy Nicklas, Eric Franz, Trey Dockendorf, and Brian L. McMichael. 2018. Open OnDemand: A web-based client portal for HPC centers. *Journal of Open Source Software* 3, 25 (2018), 622. https://doi.org/10.21105/joss.00622

[5] Meta Open Source. 2025. Docusaurus: Open Source Documentation Framework. https://docusaurus.io/. Accessed: 2025-09-05.

[6] National Science Foundation. 2024. NSF Award #2346744: CENVAL-ARC: Central Valley Accessible Research and Computational Hub. https://www.nsf.gov/awardsearch/showAward?AWD_ID=2346744 Accessed: 2025-09-06.

[7] Michael Prince. 2004. Does Active Learning Work? A Review of the Research. *Journal of Engineering Education* 93, 3 (2004), 223–231. https://doi.org/10.1002/j.2168-9830.2004.tb00809.x

[8] UC Merced Cyberinfrastructure and Research Technologies. 2025. UC Merced HPC & JupyterHub Documentation. https://ucm-it.github.io/hpc_docs/.

[9] Andy B. Yoo, Morris A. Jette, and Mark Grondona. 2003. SLURM: Simple Linux Utility for Resource Management. *Job Scheduling Strategies for Parallel Processing* (2003), 44–60. https://doi.org/10.1007/10968987_3

Building Expertise, Connections, and Communities for Computational AI and HPC Training and Education: NAIRR Pilot User Experience Group Initiatives

Nitin Sukhija
Slippery Rock University
nitin.sukhija@sru.edu

Shelley Knuth
University of Colorado, Boulder
shelley.knuth@colorado.edu

Alana Romanella
University of Colorado, Boulder
alana.romanella@colorado.edu

Marisa Brazil
Arizona State University
marisa.brazil@asu.edu

ABSTRACT

Given the rapidly changing computing landscape propelled with innovations and convergence of new cutting-edge technologies such as high-performance computing (HPC), AI, Cybersecurity, Quantum computing and more, the accelerated need for upskilling/reskilling the workforce to mitigate skills gaps is becoming increasingly important. Whether you are student, researcher, faculty, staff, or other stakeholder of academia/industry who is part of this evolving digital ecosystem, the continuous learning and adaptation of HPC along with AI best practices, research and technology is a key to remain competitive. Furthermore, a triumvirate of user expertise, connections, and communities is required to enable efficient integration of (HPC) and AI ecosystem to offer key technologies for meeting performance requirements that pushes innovations to their limits in science, engineering and other domains. To address the challenges involved in leveraging Artificial Intelligence (AI) along with computational, data, software, training, and educational resources for the U.S. research and education communities, the National Artificial Intelligence Research Resource (NAIRR) Pilot was launched in 2024. As part of this effort, the NAIRR Pilot User Experience Working Group (UEWG) have conducted various engagement initiates, such as researcher showcases, pilot industry partner showcases, webinar series, regional workshops and one national workshop on AI Training. This paper presents a reproducible roadmap based on the observations and results of the above-mentioned training and education efforts that can be used to efficiently train the next generation workforce in AI and HPC at all levels. Thus, bridging the talent gap and advancing secure and trustworthy AI in research and society.

KEYWORDS

Computational Research Tools, Artificial Intelligence, Workforce Development

https://doi.org/10.22369/issn.2153-4136/17/1/11

1 INTRODUCTION

HPC is central for empowering progress in diverse scientific and non-scientific domains. A myriad of technologies in the post petascale computing demand a significantly greater degree of parallelism than we currently observe. The rapid advancement of new HPC technologies has facilitated the convergence of Artificial Intelligence (AI), Big Data Analytics, and the HPC platforms to solve complex, large-scale, real-time analytics and applications for scientific and non-scientific fields. AI and High-Performance Computing (HPC) are complementary technologies, with HPC providing the powerful infrastructure and processing capabilities needed to run large-scale AI models, while AI enhances HPC systems by enabling intelligent data processing, pattern recognition, and optimization of complex simulations and workflows [3, 4] This synergy allows for greater efficiency, deeper data-driven insights, and the ability to tackle more complex problems than either technology could on its own, driving significant advancements across various industries from scientific research to cybersecurity. As we start the process of getting your own AI-accelerated HPC initiative running, it's important to understand common challenges:

- For AI and HPC configurations, traditionally there is a trade-off between AI and HPC requirements within the CPU architecture. AI-heavy workloads typically exchange core count for speed, while HPC workloads often prefer greater compute performance with a high core count and more core-to-core bandwidth.
- Increasingly data-intensive workloads, such as modeling, simulation, and AI, create performance bottlenecks that require solutions with high-bandwidth memory that's architected to unlock and accelerate them.
- The high level of complexity of AI in HPC is a major source of friction for adoption. The skill sets for AI and HPC are very domain specific, and finding talent skilled in both areas is difficult. However, without this talent, AI-accelerated HPC initiatives might not move forward.

Thus, as we move towards exascale, the convergent computing platforms along with a paradigm shift in the programming applications provide both challenges and opportunities, for cyberinfrastructure facilitators and educators to prepare and support a diverse community of professionals to utilize evolving HPC, equipping them to solve complex scientific, engineering, and technological problems [1, 2, 5]. In 2024, the National Artificial Intelligence Research Resource (NAIRR) Pilot was launched as a U.S. government-led public-private initiative that provides American researchers and educators with broad access to advanced AI tools, data, and computational resources. Led by the National Science Foundation (NSF), the pilot program is a stepping stone toward a permanent, full-scale NAIRR program. The pilot program is designed to democratize access to AI resources and spur innovation by addressing the following goals such as 1) Spur innovation to facilitate AI-driven research and discovery across all scientific fields. 2) Advance responsible AI to promote the development of AI that is safe, secure, trustworthy, and protects privacy and civil liberties. 3) Build talent to provide resources and training to educate the next generation of the AI workforce. 4) Expand capacity to improve the nation's overall capacity for AI research and development. As part of NAIRR pilot, the NAIRR pilot User Experience Working Group (UEWG) is involved in assessing the needs of researchers and educators using the National Artificial Intelligence Research Resource (NAIRR) Pilot. The group's findings help shape the future of the NAIRR program. The UEWG group aids in:

- Assessing user needs: The UEWG works to identify challenges and priorities for the research and education communities as they engage with AI. They use methods such as surveys and workshops to gather information.
- Informing NAIRR development: Insights from the UEWG are used to refine the pilot program and contribute to the design of the full-scale, long-term NAIRR program.
- Promoting trustworthy AI: By identifying common challenges and sharing solutions, the group fosters the development of responsible AI.
- Supporting user engagement: The group helps facilitate community engagement and support, including efforts like the Researcher Showcases, Workshops such as "AI Unlocked " workshop held in April 2025.

This paper will highlight the key findings by the UEWG summarizing key insights gathered from surveys and outreach efforts. Moreover, the paper presents a reproducible roadmap that details the mechanisms used and recommendations to address the following challenges [5]:

- Gaps in expertise: need for more high-performance computing (HPC) knowledge among users.
- Training framework: users require structured AI education, hands-on training, and workshops.
- Key training areas: such as large language models (LLMs) and general AI research applications.
- Community connections: users are highly interested in knowledge sharing and networking opportunities

This paper is coordinated by NAIRR Pilot UEWG and fosters collaborations among practitioners to explore strategies enhancing computational, data-enabled, AI and HPC educational needs. The article will discuss approaches for developing and deploying AI and HPC education and training and keeping pace with rapid technological advances: collaborative online learning tools, technology solutions supporting HPC, Accelerated Analytics, and AI applications. The paper will highlight methods for conducting effective AI and HPC education and training for emerging technologies; promote HPC and AI educators' community; disseminate best practices.

2 BACKGROUND

2.1 User Experience Working Group

The National Artificial Intelligence Research Resource (NAIRR) pilot was launched in Jan 2024 to spur innovation around AI workflows and tools and support research endeavors. Federal agencies are collaborating with government-supported and non-governmental partners to implement the Pilot as a preparatory step toward eventual full NAIRR implementation. The NAIRR User Experience Working Group, the second working group, convened as part of the NAIRR pilot works with the researchers or educators who have been selected to participate in the current open call for access to resources as part of NAIRR. Individuals selected as part of the open call are given access to resources to achieve success in their projects. These resources include not only physical hardware but also user support. The NAIRR User Experience Working Group focuses on understanding how to best facilitate the successful completion of these pilot projects. Moreover, this working group also identify the best methods to achieve these goals, pain point resolution, services or support that are needed, and other issues as they arise to formulate a successful NAIRR project after the pilot phase. The Group facilitates recommendations on good user support that enables NAIRR Pilot users to complete their work effectively and efficiently and have positive experience in the Pilot.

3 NAIRR USER EXPERIENCE GROUP INITIATIVES ROADMAP

Since the inception of NAIRR pilot, the User Experience Working Group (UEWG) has been helping to overcome the intimidation factors and barriers for current NAIRR Pilot users and potential users and to overcome issues with underutilized resources for improving the reward process. To improve the success rate for projects at the end of the pilot, the UEWG laid down an Initiatives Roadmap that involved high-level mapping out the NAIRR users' landscape and analyzing user experiences and AI/HPC programs implementations for identifying common practices and barriers resulting in framing informed recommendations to achieve the set UEWG and NAIRR pilot goals.

3.1 Map NAIRR User Landscape

To aid the NAIRR pilot current and potential users, it became of paramount importance to map the NAIRR user landscape. To map NAIRR stakeholders the first step was to identify the users, their goals, and services required to improve user experiences. However, this was not easy due to the diverse and rapidly expanding NAIRR user landscape and dynamically changing trends in user landscape. The NAIRR user landscape encompasses users who are determined by need for the massive computational power to process large datasets and to train complex AI models, for generative AI and deep learning applications and for classroom education. Moreover, the user landscape involves stakeholders ranging from hyperscale data center operators to researchers, engineers and staff from national research laboratories, universities, and government and private sector enterprises. The users of NAIRR are involved in research and innovations utilizing AI HPC systems for complex scientific workloads, accelerating discovery in various fields like genomics, molecular modeling, climate modeling, high-energy physics, autonomous driving, robotics, and manufacturing, national security, defense applications and more. Many applications and advanced users employ AI software and leverage HPC resources for tasks like data analytics, model validation, and developing new AI applications. Furthermore, with private sector dominating development of AI models in comparison to academia along with the need for AI-optimized GPUs and AI-coupled workflows for enhancing speed and performance of traditional HPC simulations, the need for better training and workforce development to ensure NAIRR users can effectively leverage these complex, highly experimental AI-HPC resources became imperative.

NAIRR pilot UEWG took many initiatives to efficiently map the landscape of NAIRR users' experience including their goals, AI tools, computing resources, and experiences to identify and gather gaps, opportunities, and interactions. The first initiative taken by NAIRR UEWG was launching the National Artificial Intelligence Research Resource (NAIRR) Pilot Project Researcher Showcases in October 2024 with an aim to share insights into cutting-edge AI projects, motivations, challenges, and engagement with the NAIRR community. In these showcases researchers were invited to present various projects involving work on digital agriculture, colorectal cancer diagnosis, neuroimage modeling for Alzheimer's risk, large foundation model pre-training, teaching AI for quality engineering, and more.

These showcases provided a unique opportunity for all users to connect with leaders in the field of AI research and education and were ideal for individuals of all skill levels who are interested in AI advancements. We had huge success with the showcases as we received more than 100 registrations for each showcase with max 187 registrations for some showcases. Furthermore, in summer 2025, UEWG launched the National Artificial Intelligence Research Resource (NAIRR) Pilot Partner Series with an aim to inform U.S. researchers and educators about the specific AI resources and services available to them at no cost through the NAIRR Pilot program's application process. The series featured in-depth presentations by the many governmental and non-governmental organizations that contribute resources to the NAIRR Pilot, such as, Databricks, Groq, Lexset, MLCommons/Croissant, Neocortex, SambaNova. Moreover, the partner series also highlighted a specific partner's resources, availability, accessibility of computing cloud platforms, tools/specialized hardware, and potential research applications.

3.2 Analyze User Experiences and AI/HPC programs Implementations

Rapid changing landscape and convergence of AI, HPC and other technologies is not only empowering innovations but also increasing gaps in specialized skills and talent pipelines. Thus, as part of the roadmap, UEWG tried to identify the training and upskilling needs of the current and potential NAIRR pilot users by identifying user experiences with respect to the different technologies present and with respect to what kind of training and skill is required to cope up with such dynamic computing environments. The group tried to analyze the experiences of users with various HPC programs implementations by trying to address various questions such as: 1) What is the NAIRR pilot user experience like? ; 2) what would potential users need to join the NAIRR pilot?; 3)What services or tool can we provide to enable the user's success?: 4) What are the needs of the current NAIRR pilot users?; 5) What are the needs of the potential users?; 6) what recommendation did the UEWG provide to best meet the needs of the users and to tailor the new user experiences?

The UEWG group took many initiatives to identify the obstacles by gathering the user input, understanding the user priorities and assessing the training needs with respect to NAIRR pilot resources. One of the most important initiatives taken by the UEWG was gathering data by administering NAIRR user experience surveys which aimed at identifying the needs and challenges faced by the AI research and education communities to aid the pilot program and inform an eventual full-scale national AI infrastructure. The survey questionnaire gathered information on various topics of interest such as the overview of AI tools, overview of research process using AI, how to get involved with NAIRR?, specialized help in certain domain sciences, preparing datasets to use AI, available computational resources, LLMs, providing participants with tangible tools to work on after workshop and walking people through the NAIRR pilot reward process.

The surveys were conducted via various mailing lists such as ACCESS mailing and NAIRR pilot user mailing lists and there were more than 402 responses recorded to the initial survey administered in Fall 2024. One of the most important findings of the survey results was the answer to the important question on the survey which was: What is your biggest challenge with AI? More than 40% of the users mentioned that their biggest challenge is: "There are too many tools, and I need help understanding what is most applicable to my domain" and almost 30% of the users mentioned that: "I am using it in my research and use some 1:1 help with specific issues."

Moreover, to compare the responses from more regional and national scales along with different levels of HPC expertise we also compared the responses of the survey respondents from the HPC community and the Rocky Mountain Advanced Computing Consortium (RMACC) community. The comparison illustrated that there is immense interest in learning about AI among communities. The members from RMACC feel more "beginner" level (only 33% of the surveys responders us AI in their research and/or projects) and members from HPC community as a whole feel more "intermediate" level (almost 62% of the surveys responders us AI in their research and/or projects). However, all community members want to learn more about tools and are interested in workforce development. Moreover, UEWG learned that AI introduces a different subset of people to the HPC community and many of them are not sure where to start. Furthermore, when asked about the most important topic of interest, more than 70% of the survey respondents voted for AI tools and their application to various domains as the topic of interest if there was an in-person training workshop organized by UEWG.

3.3 Identify Common Practices and Barriers

The NAIRR pilot encompasses many operations teams including UEWG, allocations working group, coordination group, portal and website, newsletter and metrics. After gathering the users' experience data, UEWG planned many initiatives to identify common practices and barriers. As part of the next step of the roadmap, the UEWG team worked closely together with the NAIRR Pilot Working Groups and Teams that are operating on other aspects of the NAIRR pilot process to share knowledge and insights. The goal here was to identify requirements of other working groups and resource providers who have team of user support that provides the first line of service and meet with the user support staff bi-weekly. As part of identifying pain points and solutions, the UEWG group also worked on office hours, consultations, and ticketing which aimed at in-depth support with accessing resources for AI research and advanced computing, proposal submissions and eligibility requirements, and active NAIRR Pilot awards and ACCESS resource usage. The office hours and consultations provided live, one-on-one support and guidance available for researchers and educators to access the NAIRR Pilot's AI resources. Moreover, the SCIPE awardees provide in-depth support outside of office hours or for specific issues that need tracking, users can submit a ticket.

The UEWG group also initiated the National AI Workshop in April 2025, called AI Unlocked workshop, where 768 community members applied for registration, and 304 members were invited, with 292 in attendance at the workshop event. The second series of such National AI workshop is planned in Denver again in June 2026 and will be a day and a half-long event, sponsored by ACCESS and NAIRR Pilot and University of Colorado, Boulder. In addition to the National Workshop, the UEWG group also planned and organized regional AI workshops in University of Colorado, Boulder, University of Kentucky, and University of California, Los Angeles, where more than 100 regional community members attended these regional workshops.

The goal of these AI workshops is to provide a collaborative environment and community building for existing and new AI users to discuss challenges, share experiences, and work through solutions in real-time. The workshop structure comprises two-day sessions of reproducible modules for beginners, intermediate and advanced users. The workshop includes parallel sessions providing comprehensive overview of AI fundamentals and hands-on practical experience with customizable AI tools and processes on computing resources for beginners along with providing experienced participants help in identifying and deconstructing specific challenges in their AI projects. The workshops aid in identifying common practices and barriers, uncovering skill gaps, highlighting practical barriers, addressing fears and resistance, thus promoting cross-functional collaboration and facilitating best practice sharing and a foundation for future training.

In addition to the workshops, UEWG group also submitted series of BOF's and Panels at various conferences such as SC, PEARC and EDUCAUSE that provided provides a unique opportunity to learn about the NAIRR Pilot offerings, current activities, and to discuss key hurdles for users at the intersection of HPC and AI, such as computational integration, user expectations, and effective onboarding strategies.

3.4 Develop Informed Recommendations

To overcome the identified barriers the UEWG encourages NAIRR stakeholders to incorporate informed recommendations for ensuring a more seamless, transparent, and effective experience for all users engaging with the NAIRR Pilot. As part of the roadmap, the UEWG focuses on overcoming barriers for incoming and potential users with respect to allocations, training, intimidation factors and onboarding process by providing informed recommendations, some of which are: 1) Engaging more with cutting-edge technologies for building expertise, connections, and communities; 2) Enhancing instructional design and supporting additional regional workshops that mimic the "AI Unlocked" workshop for addressing skills gap; 3) Rotating times and formats along with usability tests accommodating varying participants for AI/HPC education and training fostering redesign for usability and reproducibility; 4) Planning recurring webinars, showcases, town halls, and additional national workshops and training opportunities on topics of interest including high performance computing, PyTorch, TensorFlow, industry tools, Large Language Models, classroom and education use cases, and ethics for promoting AI/HPC awareness and gathering user feedback and pain points; and 5) Developing step-by-step guides for improving documentation accessibility and clarity for current and potential NAIRR pilot users.

The set of recommendations developed by the National Artificial Intelligence Research Resource (NAIRR) Pilot UEWG is not only intended to enhance the overall user experience of the NAIRR Pilot but also intended for the various operations teams involved in implementing several aspects of the NAIRR Pilot, particularly those with a direct connection to user experience, as well as for the federal funding agencies supporting the effort.

4 CONCLUSION

The roadmap described in this paper enabled UEWG to make a meaningful impact on user experiences by increasing access to cutting-edge infrastructure, building interdisciplinary connections, and reinforcing NAIRR's commitment to support AI research. Moreover, the roadmap outlined is an iterative, flexible, and evolving plan that aligns learning initiatives with NAIRR Pilot goals through continuous feedback and improvement in understanding of user experiences. The UEWG continues to assess and analyze user landscape and identify user needs to make recommendations for enhancing the NAIRR Pilot's operations.

ACKNOWLEDGMENTS

This work would not be possible without a great deal of support from community individuals. Lori Flora, Casey Koehler, Vikram Gazula, Karan Vahi, David Hart, and Barbara Schnell held important roles in supporting the workshop or surveys. The SCIPE and early CIP awardees are acknowledged for their contributions to office hours, training, and other tasks. Other NAIRR working groups, NAIRR Resource Providers, and ACCESS team members provided user pain points information. This work was funded in part by the National Science Foundation Office of Advanced Cyberinfrastructure, #2138286.

REFERENCES

[1] W. Filinger, J. Mullen, J. Cohen, S. Wittke, and A. Backhaus. 2024. Building HPC learning pathways: understanding our community. *Practice and Experience in Advanced Research Computing* 2024: Human Powered Computing (2024), 1–7.

[2] J. Huang, Y. Zhong, and X. Chen. 2025. Adaptive and personalized learning in STEM education using high-performance computing and artificial intelligence. *The Journal of Supercomputing* 81, 8 (2025), 981.

[3] R. K. Raj, C. J. Romanowski, S. G. Aly, B. A. Becker, J. Chen, S. Ghafoor, N. Giacaman, S. I. Gordon, C. Izu, S. Rahimi, and M. P. Robson. 2020. Toward high performance computing education. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education.* 504–505.

[4] R. K. Raj, C. J. Romanowski, J. Impagliazzo, S. G. Aly, B. A. Becker, J. Chen, S. Ghafoor, N. Giacaman, S. I. Gordon, C. Izu, and S. Rahimi. 2020. High performance computing education: Current challenges and future directions. In *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education.* 51–74.

[5] O. Terzo and J. Martinovič (Eds.). 2022. *HPC, Big Data, and AI Convergence Towards Exascale: Challenge and Vision.* CRC Press.

Investigating User Attitudes Towards and Benefits from Integrating AI Assistants into Research Computing Support

Injila Rasul
University of Massachusetts Amherst
irasul@umass.edu

Georgia Stuart
University of Massachusetts Amherst
gstuart@umass.edu

ABSTRACT

High-Performance computing clusters used for Research Computing, hosted by universities, are an essential part of the ongoing teaching, learning, and research at these institutions. Users must understand myriad scientific, mathematical, and computing concepts. They have a range of experience and comfort with these platforms, requiring regular support as they engage with it for their research. To assist users on the Unity Research Computing Platform, the support team provides the Facilitation Slack channel to get help, find relevant documentation, learn new information, and troubleshoot, requiring significant staff time and funding. This study explores the design and implementation of an AI assistant chatbot augmenting existing support with HPC Facilitator oversight. We investigate the efficacy of AI assistants in extending the productivity and impact of research computing personnel while maintaining a high degree of direct contact with users. We discuss the Human-Centered AI Design and testing process and its significance for large-scale interventions.

KEYWORDS

Human-Centered AI, High Performance Computing Facilitation, Research Computing Support

1 INTRODUCTION

The use of AI-chatbots as assistants for troubleshooting is documented across a range of platforms [4]. These AI chatbots are forms of software that simulate human-like conversations and interactions, understand user prompts, process input, and provide personalized responses [6]. AI chatbots function as digital assistants for learning and teaching, help expand user skills and abilities, and support higher-order executive skills [6].

Research computing is a field with the potential to be significantly improved by AI-assisted support. When interacting with a research computing platform, such as a High-Performance Computing (HPC) cluster, users must understand a wealth of scientific, mathematical, and computing concepts. To assist users, HPC centers employ facilitators or other support staff. However, funding enough user support staff to meet the needs of a typical HPC center is often untenable. In this study, we address initial feasibility questions around using AI assistants in research computing support, specifically in the context of the Unity Research Computing Platform (Unity) at the University of Massachusetts Amherst.

Unity is a collaborative and high-performance research computing platform that supports collaboration across multiple universities and colleges in the northeastern US. Platform users need assistance with issues ranging from simple troubleshooting, such as signing in, managing files, installing software, or accessing datasets, to complex code and workflow optimization questions. The Unity facilitation team offers help through a variety of modes, notably a Slack group. However, more basic queries may be addressed by an AI-enhanced search function, which primarily directs users to relevant areas of the documentation that can help resolve the issues that they encounter. It is imperative that the design and roll-out of an AI assistant within this system is done while being mindful of concerns such as incorrect understanding by the chatbot, user frustration, and lack of trust in AI responses [1].

The design and testing process of this AI-enhanced search/chatbot is the goal of this feasibility study. The following questions guide our work:

(1) In what ways does an AI-enhanced search for Unity users effectively facilitate basic questions that arise in research computing?
(2) What is the role that the AI chatbot can play in assisting the facilitation team in clarifying user questions and filtering out common misunderstandings?

2 RELATED WORKS

We discuss research on user involvement in design, a human-centered framework for studying AI systems, and human-AI interaction breakdowns. We describe Human-centered design in AI and discuss why it is essential to iteratively create, test, and refine the designs of such systems to minimize negative user experiences. Finally, we summarize the relevant literature on AI-assisted user experiences in education, with emphasis on the need for continued human oversight.

2.1 User-Involvement in AI Design

Prior work on integrating Artificial Intelligence into Education explores its potential impact due to its scalability. However, concerns about privacy and agency are highlighted, as end-users are often excluded from the design process [1]. Actively involving the target audience in the design of AI systems fosters trustworthiness, reliability, and a balance between human control/oversight and AI automation [2]. Incorporating end-user insights from the start of the design process helps address the identified need for greater consideration of ethical, methodical, and contextual factors [2].

https://doi.org/10.22369/issn.2153-4136/17/1/12

The viability and feasibility of AI chatbots have been investigated in studies examining dimensions of interaction in comparison with human-human interaction, yielding framework elements such as (i) words per message, (ii) words per conversation, (iii) messages per conversation, (iv) word uniqueness, and (v) use of profanity, shorthand, and emoticons [3]. User studies have also examined the framing of AI responses and their connection to aspects such as frustration, usefulness, appropriateness, and relevance in response to AI chatbot interactions [9]. Key areas of concern are breakdowns in user-LLM interactions that lead to poor user experiences, which users categorize as "irrelevant and useless output," "inability to answer," and "problematic communication style" [7].

2.2 Human-Centered AI Design Process and Principles

Human-Centered AI (HCAI) emerges when Human-Centered Design (HCD) approaches (such as centering human need, values, and perspectives) are applied to the design and development of learner analytics or AI in Education systems [5]. A key tenet is considering AI as a system that serves human interests rather than pursuing a technical goal or aiming to replace humans altogether [8].

Design Phases of the HCD process are defined as follows: Phase 1: Planning, Scoping, and Definition; Phase 2: Exploration, Synthesis, and Design Implications; Phase 3: Concept Generation and Early Prototype Iteration; Phase 4: Evaluation, Refinement, and Production; Phase 5: Launch and Monitor. This study reports results from a system in Phase 4.

Alfredo et al. outline HCAI Principles of Safety, Reliability, and Trustfulness for study design that emerged from their thematic analysis of AI system design research [1]. These principles are interconnected and do not exist in isolation from one another; rather they feed into the design process throughout and work together to foster confidence in the system and therefore its successful adaptation and use.

2.3 Safety

Data Privacy, Data Sharing, Data Collection, and Monitoring and Surveillance emerge as significant themes under the umbrella term of Safety. These elements are essential to consider when designing AI systems so that they are integrated into the DNA of the system and not transplanted in at a later stage. Data related to personal identification, academic performance, or learning progress should be considered sensitive, with the decisions about privacy influencing stakeholder willingness towards adopting the AI system. Data Sharing concerns are usually addressed by anonymizing participant identities, which allows task performance without peer judgement. For Data Collection transparency, studies needed clearer communication between researchers and participants around informed consent. For Monitoring and Surveillance, concerns around student awareness of monitoring and resulting changes in behavior are an important consideration.

2.4 Reliability

The Reliability principle is framed around system accuracy, bias, and strategies to ensure data integrity. System reliability is the accuracy and perception of the machine algorithm and human interpretation of the system. For bias, when researchers/designers are working in isolation, it increases risk of human interpretation bias to be introduced into the system and decisions around the design of the system. Therefore, including diverse perspectives, ideally from including end-users, actively in the design process increases reliability of the system.

2.5 Trustfulness

Trustfulness in the system can refer to stakeholder trust, user perception, and accountability. Trust in human-centered AIED systems is improved by providing an explanation when users request feedback and accuracy of the information it provides. Additionally, it connects to transparency around the way the system gathers and manages information and communicates or documents research.

2.6 Human-in-the-Loop by Design in Self-Regulated Learning

Research around AI in Self-Regulated Learning (SRL) is in its infancy [4]. SRL is very nuanced, with conceptual models and frameworks drawing from different aspects of behavioral, cognitive, and affective theories of learning. Notably, Zimmerman's (2002) SRL model has three phases: Forethought (i.e., goal setting, strategic planning), Performance (i.e., help-seeking, learning monitoring, and instructional strategies), and Self-Evaluation (i.e., self-reflection, feedback) [10]. Therefore, AI applied in this domain will also need a broad and deep capacity for augmenting and supporting SRL.

The distinction between human-centered self-regulation with AI as facilitator (a tool to monitor, control, and support learning) and AI-centered self-regulation with AI making data-based decisions is an important one to make cite4. Most AI applications engage target-specific dimensions of the SRL cycle rather than the whole process holistically and thus support some aspects more than others.

2.7 Breakdowns in User-AI Interaction

Errors and warnings arise in user-AI interactions when user requests cannot be fulfilled due to technical limitations or policies restricting specific responses [9]. When these breakdowns occur in conversations, users might repeat or clarify, emulating a human-human interaction, but for a chatbot interaction, that might not lead to a clarification but rather frustration. Handling user frustration in such instances is crucial for managing these interactions effectively. Wester et al. [9] tested methods to manage the HCI in the case of request denials and found that avoiding baseline denials that do not adequately inform the user as to why their request is not answered. Instead, diverting denial styles were more favorable, which navigates the denial by suggesting alternatives that the AI can assist with [9].

3 METHODOLOGY

For the context of this research, "effective facilitation" is operationally defined as users perceiving themselves as more capable of solving problems with the tool in use than without it. The bespoke instruments were piloted and refined using feedback from the Unity facilitation team, which consists of experts in optimizing High Performance Computing workflows and designing learning technology.

Table 1: Pre and Post-Test Questionnaire

	QUESTION COMPONENTS
LEARNER PROFILE	1. I am a frequent user of research computing clusters (for example, Unity). 2. I am comfortable using research computing clusters to conduct my research. 3. I consider myself well-versed in solving computing issues. 4. When I face an issue, I can comfortably navigate the internet or documentation to help solve my issue. 5. I am comfortable navigating through software documentation to resolve issues I face. 6. When I feel stuck in a platform I am using, I utilize any available AI-assistance. 7. I prefer AI-assistance to human-assistance for questions around computing/software. 8. If I get stuck on a technical problem during research computing, there is no chance I'll figure it out on my own.
AI-USE	1. I am not comfortable prompting AI chatbots for assistance when I am stuck on a technical problem. 2. If I get stuck on a technical problem during research computing, there is no chance I'll figure it out on my own. 3. There is usually only one correct approach to solving a technical issue while working on research computing. 4. AI chatbots are confusing to use. 5. The instructions that AI-chatbots give are clear or easy to follow. 6. When I use AI chatbots to problem-solve, I feel frustrated if the answer is unclear to me. 7. AI chatbots can give incorrect and misleading answers to my question prompts. 8. AI chatbots are useful for preliminary assistance for simpler problems. 9. I prefer human assistance when the problems are more complex or multi-faceted. 10. I am better able to solve problems when I have a human assisting me. 11. I prefer human assistance when I am facing software/computing issues, compared to AI assistance.
COGNITIVE WALKTHROUGH	The participant will be given a series of tasks, one at a time, to complete on the Unity HPC Cluster. Once a task is completed to the satisfaction of the participant, the researcher observer will give them the next task. Tasks are intentionally open-ended, but can be answered from existing documentation. Tasks are roughly ordered by difficulty. Task list (example): 1. Log into Unity OnDemand and upload a test file to your home directory (test file is an ipynb file for subsequent tasks) in a subdirectory named "Unity ai_study". 2. Start a JupyterLab session with 4 CPU cores and 4 GB of Memory for a 1-hour time limit in the "cpu" partition. 3. Open the notebook (ipynb) file uploaded in Task A with the default Python kernel and execute all cells. 4. Create a non-interactive job to run the notebook (ipynb) uploaded in Task A without the JupyterLab Unity OnDemand app.
INTERVIEW: SEMI-STRUCTURED QUESTIONS	1. Which tasks, if any, were you familiar with in advance? 2. In general, what is your opinion on AI-chatbots being used to assist in searching for answers or clarification? 3. In general, what is it like for you to interact with an AI-chatbot? 4. Can you think of a time when you felt frustrated while using an AI-chatbot? 5. Overall, would you consider AI-chatbots to be useful in your experience, and why? 6. In your experience, have AI-chatbot responses typically been appropriate to the situation and/or your question? 7. In your experience, was the content of the AI-chatbots' answer clear and relevant to your question? 8. Have you asked questions in the user Slack before? 9. If YES, in what ways was your experience different? Was it better or was it worse? 10. If NO, what are the reasons why you have not used the user facilitation slack channel?

3.1 Data Types and Collection

There are four categories of data collected for this project. The collection and analysis of data is currently ongoing. Instrument components are given in Table 1.

(1) Pre- and Post-test (qualitative in nature) taken by users on user experience and self-perception of computing ability, administered through Qualtrics.
(2) One-on-One semi-structured interviews with users on user experience
   (a) Cognitive walkthrough exercises,
   (b) Experience-related questions.
(3) Pre- and Post-test (quantitative in nature) taken by the user on content knowledge before and after the bot facilitation experience, administered through Qualtrics.
(4) Qualitative analysis of the question-and-answer log from the active Unity help-desk channel.

3.2 Participants

Our target audience is the existing Unity users, typically research teams utilizing the cluster for their research projects. This includes PIs, research associates, and student research assistants who are using the system for their active research projects. User consent is obtained at the start of the testing regarding the use of their data for developing and refining the design of the AI chatbot.

3.3 Data Analysis

The collected data is analyzed for themes around user experience, the degree of facilitation the AI-chatbot provides, and the ways in which it can improve. The qualitative coding process involves classifying major and minor themes that emerge from the data, as well as developing a codebook. The initial codebook uses the HCAI framework of themes around Safety, Reliability, and Trustfulness.

4 PRELIMINARY RESULTS

Data collection and analysis are currently ongoing and scheduled to conclude by early Fall of 2025. Data analysis is an iterative process conducted in collaboration with Unity team facilitators and administrators, who participate in work sessions and presentations through which emergent and key findings and patterns are discussed. Experts in high-performance computing and computational science education raised questions, shared insights, and surfaced contextual knowledge that helped shape the ongoing AI assistant development and strategies for data collection and analyses. In line with the centers' goals, decisions about which analyses to conduct and what data would be most useful were made by researchers.

Discussions with the HPC facilitators and review of the existing Slack channel messages indicated a level of redundancy in existing questions, prior to deployment of the AI chatbot. The facilitation team found that the developed AI assistant reasonably answered many of the common questions. However, data collection from participants who are not subject matter experts is forthcoming. The facilitator panel further provided feedback to refine the tasks for the cognitive walkthrough to cover tasks at a range of complexity, shown in Table 1.

5 SIGNIFICANCE

Rolling out the AI-chatbot to the entire audience of researchers at the six different institutes that use this research cluster and eventually integrating it within the User Facilitation Slack is a large-scale change to the system and requires a learner-focused approach to its design and implementation. The study focuses on ensuring AI is integrated intentionally and intelligently, not for the sake of an AI tool but rather with the intention of centering learner needs and utility. A human-centered approach to developing and deploying new AI tools ensures they are tailored to address gaps in human-provided support and support humans who are the lead facilitators.

Retaining the Human-in-the-Loop also ensures that learner frustration is managed, learners are understood and supported in their work, and that custom facilitation and support are readily available. The automation of facilitation in the research computing environment is essential; however, exercising caution around the design, development, and large-scale deployment of such a tool is necessary to be mindful of the limitations of such technology. The bespoke instrument we developed provides fellow researchers with a starting point for assessing learner experience, identifying pain points, and addressing the issues that are identified. Positioning the AI tools in the facilitator role, while incorporating ongoing monitoring and evaluation by the human ensures that the end-user's learning is managed while lowering the teaching burden on the team.

Ultimately, this design approach strengthens the process for scaling high-quality learner facilitation and learning in a more informal SRL context. It reiterates that AI-enhanced learning technology tools should be developed through an iterative process, driven by existing and evolving literature on design principles, and be contextually intelligent by integrating the needs of the end-learners from the start.

ACKNOWLEDGEMENTS

Our thanks to NSF for supporting this research effort [Award Number: 2430001, Project Title: CC* Compute-Region: Harmony for Research Computing: A Green Compute Solution Designed to Enable Emerging Workflows and to Maximize Access for New England Researchers and Educators].

REFERENCES

[1] R. Alfredo, V. Echeverria, Y. Jin, L. Yan, Z. Swiecki, D. Gašević, and R. Martinez-Maldonado. 2024. *Human-centred learning analytics and AI in education: A systematic literature review.* Artificial Intelligence, 6, 100215, Computers and Education. https://doi.org/10.1016/j.caeai.2024.100215

[2] M. Bond, H. Khosravi, M. De Laat, N. Bergdahl, V. Negrea, E. Oxley, P. Pham, S. W. Chong, and G. Siemens. 2024. A meta systematic review of artificial intelligence in higher education: A call for increased ethics, collaboration, and rigour. *International Journal of Educational Technology in Higher Education* 21, 1 (2024), 4. https://doi.org/10.1186/s41239-023-00436-z

[3] J. Hill, W. Randolph Ford, and I. G. Farreras. 2015. Real conversations with artificial intelligence: A comparison between human–human online conversations and human–chatbot conversations. *Computers in Human Behavior* 49 (2015), 245–250. https://doi.org/10.1016/j.chb.2015.02.026

[4] M. Lan and X. Zhou. 2025. A qualitative systematic review on AI empowered self-regulated learning in higher education. *Npj Science of Learning* 10, 1 (2025), 21. https://doi.org/10.1038/s41539-025-00319-0

[5] C. Lang and L. Davis. 2023. Learning Analytics and Stakeholder Inclusion: What do We Mean When We Say "Human-Centered"?. In *LAK23*. 411–417. https://doi.org/10.1145/3576050.3576110

[6] P. Pergantis, V. Bamicha, C. Skianis, and A. Drigas. 2025. AI Chatbots and Cognitive Control: Enhancing Executive Functions Through Chatbot Interactions: A Systematic Review. *Brain Sciences* 15, 1 (2025). https://doi.org/10.3390/brainsci15010047 Article 1.

[7] M. Skjuve, A. Følstad, and P. B. Brandtzaeg. 2023. The User Experience of ChatGPT: Findings from a Questionnaire Study of Early Users. In *Proceedings of the 5th International Conference on Conversational User Interfaces*. 1–10. https://doi.org/10.1145/3571884.3597144

[8] U. A. Usmani, A. Happonen, and J. Watada. 2023. Human-Centered Artificial Intelligence: Designing for User Empowerment and Ethical Considerations. In *5th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. https://doi.org/10.1109/HORA58378.2023.10156761

[9] J. Wester, T. Schrills, H. Pohl, and N. van Berkel. 2024. As an AI language model, I cannot. In *Investigating LLM Denials of User Requests. Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–14. https://doi.org/10.1145/3613904.3642135

[10] B. J. Zimmerman. 2002. Becoming a self-regulated learner: An overview. *Theory into practice* 41, 2 (2002), 64–70.

March 2026

Volume 17 Issue 1

ISSN 2153-4136 (online)