2023

A Framework For Teaching Machine Learning For Engineers

Lauren SINGELMANN
Minnesota State University, Mankato, United States of America, lauren.singelmann@mnsu.edu

Jacob COVARRUBIAS
Minnesota State University, Mankato, United States of America, jacob.covarrubias@mnsu.edu

A CONCEPTUAL FRAMEWORK FOR TEACHING MACHINE LEARNING FOR ENGINEERS (PRACTICE)

L. Singelmann [1]
Minnesota State University, Mankato – Department of Integrated Engineering
Virginia, MN
ORCID: 0000-0003-3586-4266

J. Covarrubias
Minnesota State University, Mankato – Department of Integrated Engineering
Virginia, MN

Conference Key Areas: *Education about an education with Artificial Intelligence; Curriculum Development*
Keywords: *Machine learning, artificial intelligence, curriculum*

ABSTRACT

As machine learning and artificial intelligence become increasingly prevalent in our day-to-day lives, there becomes an even greater need for literacy in machine learning for those outside of the computer science domain. This work proposes a conceptual framework for teaching machine learning to engineering students with the goal of developing the knowledge and skills needed to apply machine learning techniques to engineering problems.

Many machine learning courses in computer science, math, and statistics focus on the theoretical basis of machine learning algorithms and assessment. This framework takes a fundamentally different approach by creating a course structure for machine learning practitioners rather than machine learning developers.

The presented framework breaks machine learning into four fundamental principles that should be used in any machine learning solution: data (what information we can use to develop our solution), task (what we are trying to accomplish with our solution),

---
[1] *L. Singelmann. Email: lauren.singelmann@mnsu.edu*

algorithms (what computational models we are using to create our solution), and assessment (how we are measuring the success of our solution). To teach this framework, the structure of the course includes creating concept maps of the four fundamental principles and relevant topics, completing coding tutorials, and creating in-class presentations that use and apply the four fundamental principles.

The paper will present the need for machine learning and artificial intelligence education within engineering, the framework and supporting learning theory, suggested activities for implementation, and lessons learned from the implementation of this framework in a 1-credit course for engineering students.

1 INTRODUCTION

Machine learning and artificial intelligence (AI) are becoming increasingly embedded in the industries that drive our world -- including healthcare, energy, infrastructure, marketing, and education. The World Economic Forum's Future of Jobs Report 2020 demonstrates the growth of AI through its survey of hundreds of companies in 26 countries. Its list of 20 emerging job roles includes titles such as data analysts and scientists, AI and machine learning specialists, big data specialists, and digital transformation specialists. The report also illustrates the importance of engineers as society make this transformation into new jobs and roles; jobs traditionally in the "engineering" sector will soon shuffle into new emerging industry sectors including "cloud computing" and "data and AI". Although these spaces will be shared by those in sectors such as information technology, engineers will play an integral role in the emergence of these new areas of industry (World Economic Forum 2020). To support future engineers entering these roles, there is a greater need for structured spaces for learning about artificial intelligence and machine learning in engineering classroom settings.

A variety of engineering programs have begun offering machine learning courses, but there has been little published about how machine learning for engineers should look different from other computer science-focused machine learning courses. Engineering and computer science courses in machine learning should have different goals and therefore different structures; rather than developing expertise in the theoretical basis of machine learning algorithms, engineering courses in machine learning should be focused on developing practical and applied machine learning knowledge and skills. To fill this need, a conceptual framework for teaching machine learning to engineers was created. This publication shares the research basis for the conceptual framework and its creation, introduces the conceptual framework, describes the implementation of the framework in a one-credit course, and shares lessons learned from implementation.

2 THE BASIS FOR A CONCEPTUAL FRAMEWORK

Whereas computer science courses in machine learning help students develop expertise through a deep exploration of theory and practice, we propose that engineering courses in machine learning should serve as a "shortcut" to developing expert-like thinking in the topic. Rather than serving as theoretical experts in machine learning, engineers serve as practical experts in machine learning. To support this practical development, the conceptual framework was designed considering learning sciences research about the process of building expert-like thinking. In the seminal report *How People Learn*, three components of building expert-like thinking were identified: 1) gaining a foundation of factual knowledge, 2) understanding these facts and ideas in the context of a conceptual framework, and 3) organizing knowledge in ways that facilitate retrieval and application (National Research Council 1999). All three of these components are important in helping learners develop competency in machine learning; they must gain a foundation in the key principles of machine learning, but they also must have a conceptual framework that helps them organize those ideas and apply them to new situations. This does not mean that students take the course to become machine learning experts. Rather, they take the course to structure and organize their fundamental knowledge like an expert so they can draw on that knowledge in practice.

The factual knowledge and means of facilitating retrieval and application can be adjusted depending on the context of the course, instructor, and students, but the proposed conceptual framework serves as a unified grounding for teaching and learning machine learning as engineers.

3 METHODOLOGY

The conceptual framework was developed by reviewing existing textbooks in machine learning (Bishop and Nasser 2006; Witten and Frank 2002; Mohri et al. 2018; Alpaydin 2020; Shalev-Shwartz and Ben-David 2014) and identifying popular concepts and topics across the textbooks. These topics were then grouped using an inductive coding approach with the goal of creating a conceptual framework for teaching and learning of machine learning for engineering students. The developed conceptual framework was then implemented in a pilot course, and each of the authors (one instructor and one student) provided perspective and insight on the use of the framework within the course.

4 THE DEVELOPED CONCEPTUAL FRAMEWORK

Through the analysis of the various machine learning textbooks, four key fundamental principles of machine learning were identified: data, tasks, algorithms, and assessment. Table 1 presents the four fundamental principles of the conceptual framework and their definitions, as well as example concepts from the textbooks that apply to each of the fundamental principles.

In the context of the three components of building expert-like thinking introduced in Section 2, this framework supports both the gaining of factual knowledge (such as the concepts listed in the table) and the organization of that knowledge for the purpose of application.

For example, one engineering application of machine learning is design optimization. One of the goals of the framework is to help students bridge the gap between what they are learning in class and a specific application. Using the framework, a student in the course could review a paper that discusses how neural networks have been used to optimize turbine blade aerodynamics (Zhang and Janeway 2022). Although the work presented includes significant technical depth, the framework can still be applied by someone new to the field of machine learning to summarize the study and connect their factual knowledge from class to a specific application. *Data* from the study includes 20 quantitative blade design parameters. The *task* being performed is regression to predict isentropic efficiency and power output from the blade design parameters. The *algorithm* chosen was an artificial neural network trained with various blade designs with known performance, and the resulting model was assessed by comparing the known performance metrics to the predicted performance metrics using evaluation error percentages. Although a student may not understand many of the technical details of the paper, they are able to understand the goal of the work and if that goal was met. On the learning front, they are further strengthening their understanding of factual knowledge by applying their conceptual framework to a practical application, ultimately building expert-like thinking of the subject.

Table 1. The Proposed Conceptual Framework for Teaching Machine Learning for Engineers. The framework consists of four fundamental principles.

Fundamental principle	Definition	Example concepts
Data	The characteristics of data and the processes used to utilize that data for machine learning processes	Types of data: numerical, categorical, time series, text Data vocabulary: targets, classes, and features
Tasks	The goal of the machine learning solution	Overarching task categories: supervised, unsupervised, semi-supervised learning Specific tasks: classification, regression, clustering, association analysis
Algorithms	The mathematical, statistical, and/or computational approach to completing the machine learning task	Example algorithms: support vector machines, decision trees, logistic regression, neural networks, a priori, k-nearest neighbors, DBSCAN
Assessment	Metrics for quantitatively assessing the ability of the machine learning model to complete the task	Example assessment metrics: confusion matrices, accuracy, recall, precision, mean absolute error, lift, support, Davies-Bouldin Index

5 EXAMPLE IMPLEMENTATION OF THE FRAMEWORK IN A COURSE

The conceptual framework was applied to a 1-credit machine learning elective course for junior and senior-level students majoring in Integrated Engineering. Students came from two separate programs within the same department. One of the programs is a work-based engineering program where students spend their last four semesters of the program working full-time in engineering co-ops while taking their courses in the evening. The second program is a project-based engineering program where students work on industry-sponsored projects.

The information presented in this section serves as a single example of how the framework could be implemented in a course; others looking to implement the framework should consider the concepts and activities that best fit their student and program needs.

5.1 Structure of the Course

This course was taught using a flipped classroom approach, meaning students watched videos about the course content before coming to class, and class time was used for activities and discussion. The class size was 12 students.

Table 2. Modules and example topics introduced in a 1-Credit Implementation of the framework. The first module introduces the conceptual framework that will be used in the course, as well as basic concepts related to data and tasks. Modules 2-5 each dive deeper into the 4 tasks and cover the four fundamental principles in the context of each of the tasks. The final module is a review of concepts covered.

Module	Topics introduced
1. Introduction to the Fundamental Principles	Machine learning, data, tasks, algorithms, assessment, target, features, types of data (numerical, categorical, time series, text), supervised and unsupervised tasks, classification, regression, clustering, association analysis
2. Classification	Classes, training set, testing set, balanced datasets, unbalanced datasets, oversampling, undersampling, support vector machines, neural networks, dummy classifiers, confusion matrices, accuracy, precision, recall, F1 score
3. Regression	Underfitting, overfitting, linear regression, polynomial regression, Ridge/LASSO, dummy regressors, mean absolute error, mean squared error, root mean squared error, R squared
4. Clustering	Dimensionality reduction, principal component analysis, centroid-based clustering, density-based clustering, hierarchical clustering, k-means clustering, DBSCAN clustering, agglomerative clustering, internal evaluation, external evaluation, Davies-Bouldin Index

5. Association Analysis	Association rules, antecedent, consequent, a priori algorithm, FP growth algorithm, support, confidence, and lift
6. Review and Wrap-Up	Review of previously covered topics

5.2 Example Course Activities

Throughout the course, seven main deliverables were introduced. Four of the deliverables were due weekly and served as formative assessment: the concept map activity, the coding tutorials, the learning journals, and class engagement. Three of the deliverables were due at the end of the course and served as summative assessment: the in-class presentation, the deep learning activity, and the final verbal exam. The formative assessments focused on the first component of developing expert-like thinking: gaining a foundation of factual knowledge. The summative assessments focused on the third component of developing expert-like thinking: organizing knowledge in ways that facilitate retrieval and application. The four fundamental principles were used in all aspects of the course to promote the second component of developing expert-like thinking: understanding knowledge in the context of a conceptual framework.

<u>Concept Map Activity:</u> Each week, 10- to 20-minute videos were posted covering the topics presented in Table 2. To show engagement with the videos, students were asked to put each of the topics on a concept map – a visual representation of how ideas are connected. They were instructed to include descriptions and/or images of the required topics, questions that they had, and at least 2 concepts that were *not* covered in the videos. For example, during the classification module, all students were instructed to add the topics in Table 2, but they were also given additional topic ideas such as other algorithms (random forest, K-Nearest neighbors), other performance metrics (log loss, ROC AUC), or other considerations of classification models (binary vs. multi-class vs. multi-label); students could choose to add these ideas as their additional topics or identify their own. The concept map was designed to facilitate the organization of the factual knowledge that students were gaining throughout the course. It also encouraged self-directed learning by requiring that students add concepts other than the ones covered in the videos.

<u>Coding Tutorials:</u> During the course, students completed two coding tutorials in Python. For the first activity, students created a classification algorithm that predicted part failure using a variety of quantitative features. For the second activity, students created a clustering algorithm that grouped unlabelled hand-written images into clusters; they assessed the clustering algorithm by using external evaluation to see how often images of the same number were clustered together. These tutorials were created by the author using Replit, a collaborative web-based integrated development environment. Because this course is focused on using machine learning as a tool rather than developing coding expertise, all relevant functions were provided to students to use. In addition, students could access fully functioning code from the instructor if they got stuck. Rather than being assessed on their ability to write code, they were assessed on their responses to questions that were embedded

in the activities. These questions were related to each of the fundamental principles. Example questions included "What type of data are each of the input features?", "How does your code account for the fact that there are more samples that do not fail than samples that fail?", and "Which of the models perform the best? Use your assessment results as evidence."

Learning Journals: Each week, students were given a prompt to reflect on in 1-2 paragraphs. These prompts varied in topics including reflection on how they practiced self-directed learning while working on the coding tutorials, ethical considerations of machine learning, and how they see machine learning applying to their area of engineering.

Class Engagement: Because content was delivered during the videos that students watched outside of class, in-class time was used for activities, discussion, and in-class presentations. Students were assessed on their participation in these activities. If they were not able to attend class, they watched a recording of the class and submitted a reflection of what their takeaways were to demonstrate engagement with the material.

In-Class Presentation: One of the additional goals when developing the pilot course was giving students the chance to explore relevant and current applications of machine learning. Students worked in groups to create a presentation and corresponding activity about a topic of their choice. Each group was given 25 minutes during class time to share their work and lead discussion.

Deep Learning Activity: The deep learning activity was a summative assessment where students were asked to connect the course concepts to an engineering application of machine learning. They could write a paper or create a video about their application, and they were assessed on four criteria: 1) their ability to identify and describe an engineering application of machine learning, 2) their ability to apply the four fundamental principles and course concepts to their application, 3) their ability to show technical depth beyond what was covered in the course, and 4) their professional communication (including citing relevant academic sources).

Final Verbal Exam: The final verbal exam was a final opportunity for the instructor and students to discuss the course concepts one-on-one. Students were instructed to come with an engineering application of machine learning, and the instructor could ask any question about the application related to the topics included in Table 2. Many students chose the same application that they covered in their deep learning activity, but the verbal exam allowed for a space where the instructor could further probe student understanding of the course concepts and help clarify any final misconceptions.

6 STUDENT PERSPECTIVE

The narrative below is written by a student who was enrolled in the course.

I initially took this course because I didn't know what machine learning was or how to use it. Still, as an engineering student who loves building my toolbelt, I felt like it *would be a great opportunity to learn about a topic outside of my mechanical engineering focus so that I could bring more value as a future engineer. After taking the course, I gained confidence in applying machine learning algorithms as learned through the coding tutorials, identifying scenarios where machine learning algorithms add value to a project, and selecting appropriate machine learning algorithms based on my data and goal.*

As part of my program, I have the opportunity to work full-time in industry as an engineer while completing coursework towards my degree. At my current company, a defense contractor in Southern California, we had a data collection device that utilizes a regression model to predict a characteristic of our products, but it wasn't working well. Through this course, with the knowledge I gained, I was able to troubleshoot the issue with a recently hired data engineer. With my expertise in the product and the value of the project, and the data engineer's technical perspective, I was able to utilize the fundamental principles learned in this course as the foundation to ask him the right questions and provide him with the required information for us to successfully troubleshoot and improve the machine learning model. It was a great feeling to see the model finally produce more accurate results and to know that my education immediately applied to a real problem, allowing me to add value to my company.

7 INSTRUCTOR PERSPECTIVE

The instructor offers the following takeaways from implementing the framework:

1. Even if a student is choosing to take the course, they may not have any background in machine learning. In their reflections, many students noted that they came into the class knowing little to nothing about machine learning. There is benefit in spending sufficient time at the beginning of the course to help the class develop a shared definition of machine learning (while recognizing that even experts have various perceptions of what is and is not machine learning).
2. Students appreciated the open-source approach to coding. With an abundance of code available on the internet, being able to understand and adapt someone else's code can be just as beneficial of a skill as writing your own code. Although writing code is a valuable skill, the coding activities in this course had a fundamentally different goal; they served as a space for students to better understand how the fundamental principles and other course concepts are integrated into a coding solution in an application that is relatively simple, but still relevant.
3. Vocabulary can be a challenge, so it is important to keep discussion open about how the instructor does and does not define terms. For example, words like "task", "precision", and "unsupervised learning" have very specific meanings in the context of machine learning, but students may come in with other ideas of what these words mean. "Precision" in the context of classification assessment refers to the proportion of positive cases that were predicted to be positive. However, students may hear "precision" and think about the consistency of an algorithm more generally.

4. Students left with positive feelings about their ability to understand and work with machine learning tools and applications. Activities like the concept map and the deep learning activity helped them realize that there is more to be learned, but they remained confident in their ability to ask the right questions and navigate engineering applications of machine learning.

8 SUMMARY

This paper presented a conceptual framework for teaching machine learning to engineering students. The development of the framework combined theory and practice by 1) employing learning theory about gaining expert-like thinking practices to design the structure of the course and 2) analyzing existing machine learning courses and textbooks to determine the content that should be covered. The conceptual framework included four fundamental principles: data, tasks, algorithms, and assessment. All course concepts and activities were framed around these fundamental principles. This helped students develop expert-like thinking about machine learning topics and an ability to understand, discuss, and work with engineering applications of machine learning.

REFERENCES

Alpaydin, Ethem. *Introduction to Machine Learning*. MIT press, 2020.

Bishop, Christopher M., and Nasser M. Nasrabadi. *Pattern Recognition and Machine Learning*. Vol. 4. Springer, 2006.

Mohri, Mehryar, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT press, 2018.

National Research Council. *How People Learn: Bridging Research and Practice*. National Academies Press, 1999.

Shalev-Shwartz, Shai, and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge university press, 2014.

Witten, Ian H., and Eibe Frank. ‘Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations’. *Acm Sigmod Record* 31, no. 1 (2002): 76–77.

World Economic Forum. ‘The Future of Jobs Report 2020’, 2020.

Zhang, Chao, and Matthew Janeway. "Optimization of Turbine Blade Aerodynamic Designs Using CFD and Neural Network Models." *International Journal of Turbomachinery, Propulsion and Power* 7, no. 3 (2022): 20.