DASC 41103
Machine Learning
Project 3
Applying CNN

**Total Points**: 200

**Lesson Objective Alignement**:

- Design and implement machine learning models using Python.
- Evaluate model performance.
- Work with real-world datasets and apply feature preprocessing.
- Communicate machine learning principles and methods to diverse audiences.

**Project Objective**: Demonstrate ability to apply CNN in Python with PyTorch.

**Description**: The University of Arkansas has found several venders on Etsy using the officially licensed Razorback logo without permission or a licensed agreement. Therefore, they hired your team to create an image classification model to classify if the images contain the official licensed logo or not. This will help them identify venders to investigate in the future to prevent revenue loss.

For this project, students will work in **pairs (groups of 2)** to complete each part, provide requested deliverables, and answer any questions.

**Data:** Self-identified and created based on images found on Etsy

- The official logo can be found at https://brand.uark.edu/graphic-identity/logos-and-wordmarks/the-razorback.php.
    - This should be one of the images you use in your dataset.
- At least 50 pictures, which contain images with and without the official razorback.
    - Split is determined by you and will impact performance.
    - I would be selective of my images to make it easier for my model to find the official logo. For example, I would stay away from using images that contain major modifications. This would be helpful in a real project but will make it harder for your model to train on a small dataset.
    - Use key words, such as “razorback” and “Arkansas razorback” at https://www.etsy.com/

**Parts:**

1. Dataset Development
    a. Developing a picture dataset of **at least** 50 images based on the guidance above. This requires finding the images and creating the dataset of the image data with the correct classification.
    b. Load completed dataset for ingestion into your CNN model.

    c. Preprocess dataset with final transformation in your data transformation pipeline to resize images to be 500 by 500.
2. CNN
    a. Load the dataset.
    b. Create CNN architecture.
    c. Train and test CNN model on dataset while
    d. Tune parameters to develop multiple candidate solutions.
    e. Evaluate models using appropriate metrics.
    f. Select most promising model.
        i. Save full model using the following name:
            1. Group_#_CNN_FullModel.ph
3. Reflection and Conceptual Questions
    a. How did you create your dataset and determine split of official logo versus not official logo?
    b. Why did you choose the specific architecture for the final model?
    c. How did you monitor and mitigate overfitting?
    d. What future efforts do you recommend to improve model performance?

**Deliverables**:
*Students should submit a video recording of their presentation, their slides, saved full model file, and a link to their open GitHub repository via blackboard.*

- 15-minute presentation that at a high-level discussing what you did in part 1 and 2 and the answers to questions in part 3 in more depth.
- 1 saved model file. We will use this model to predict a classification of 1 (i.e., official logo) or 0 (i.e., not official logo) on new unseen data.
- Linke to your group's open GitHub repository with code that is clean and well-commented.

**Grading**:
All team members will receive the same grade unless a team member requests otherwise. Grades will be based on the grading rubric below.

## Project Grading Rubric

| **Group Number:** | Points |
|---|---|
| 1. **Presentation:** | |
| **Project Discussion:**<br>**30 points**: A clear, concise, and high-level summary of the methodology, challenges, and results for Part 1 and 2. Demonstrates a strong understanding of the project's practical steps.<br>**15 points**: Provides a partial summary of the project parts. Some key steps or findings are missing or unclear.<br>**0 points**: The project overview is missing or lacks a coherent narrative. | 30 |

| | |
|---|---|
| **Conceptual Depth:**<br>**30 points**: Offers a comprehensive and articulate explanation of the concepts from Part 3, demonstrating a deep understanding of algorithms and theory.<br>**15 points**: Provides a basic but incomplete explanation of the concepts. Lacks depth or contains some inaccuracies.<br>**0 points**: The conceptual answers are missing or incorrect. | 30 |
| **Presentation Quality:**<br>**20 points**: The presentation is well-structured, professional, and within the 15-minute time limit. Visuals on the slides are clean and easy to follow. Audio and video quality are excellent.<br>**10 points**: The presentation is somewhat disorganized or exceeds the time limit. Some visuals are cluttered, or the audio/video quality is poor.<br>**0 points**: The presentation is incomprehensible, significantly exceeds the time limit, or was not submitted. | 20 |
| **2. Code and Repository** | |
| **Code Functionality:**<br>**30 points**: All code runs without errors, producing the expected outputs and plots.<br>**15 points**: The code runs with minor errors or does not fully complete all tasks.<br>**0 points**: The code fails to run or contains significant errors, making it unusable. | 30 |
| **Code Quality and Comments:**<br>**30 points**: The code is clean, logically organized, and highly readable. It includes comprehensive and helpful comments explaining key functions, logical blocks, and complex steps. Variable names are descriptive.<br>**15 points**: The code is functional but could be cleaner. Comments are sparse or do not adequately explain the logic.<br>**0 points**: The code is unreadable, not commented, or disorganized. | 30 |
| **GitHub Repository:**<br>**20 points**: The repository is public, contains all required code and a README file, and is easily accessible via the link provided. The commit history is logical.<br>**10 points**: The repository is missing a README, is not public, or is difficult to navigate.<br>**0 points**: The GitHub link is broken, or the repository is not submitted. | 20 |
| **3. Model Performance and Deliverable** | |
| **Model Output File Naming & Submission:**<br>**10 points**: The model output file is submitted to Blackboard with the correct and exact file name as specified in the assignment.<br>**5 points**: File is submitted but some have incorrect name or is missing.<br>**0 points**: No files submitted. | 10 |
| **Model Accuracy on Validation Set:** Based on class performance!<br>**30 points**: Your models achieve an acceptable level of accuracy on the provided validation set, demonstrating a robust implementation and an effective approach to feature preprocessing and model training.<br>**15 points**: Your models achieve a moderate level of accuracy.<br>**0 points**: Your models achieve a low level of accuracy, indicating fundamental errors in the implementation or training process. | 30 |
| **Total** | |