CSCI 578 Machine Learning
Competition 2022: Butterfly Classification

Instructor: Murat Dundar

Due date: 4/10/2022, 11:59pm

Please type in your assignment and submit it online through Canvas. No hard copies will be accepted.

Task 2: Multiple-instance, Closed-set Butterfly Classification using DiNo Features

Dataset: The train, validation, and test splits are the same as Task 1. There are two important changes. First, unlike Res-Net 101, which is pretrained in a supervised fashion using 1M images from 1K classes in ImageNet, DiNo is pretrained on the same data by self-supervised learning without using any class labels. ~~This pretraining phase is followed by fine-tuning on a Butterfly dataset containing over 20K images, a subset of which is used for this competition.~~ The dimensionality of the DiNo feature embeddings (384) is not only substantially lower than Res-Net 101 (2048) embeddings but also expected to have superior generalizability thanks to its self-supervised training process that eliminates the class bias. Second, in addition to feature embeddings obtained from the whole images (stored in $whole.mat$), you are provided additional embedding vectors extracted from nine different parts of each image (stored in $parts.mat$). The content of $whole.mat$ and $parts.mat$ files are described in Tables 1 and 2. 2D Part IDs are explained in Figure 1.

Goal: The goal of the second task is to predict species labels for testing samples. For every sample id in the $test\_sampleid$ variable in the $whole.mat$ file you will submit one prediction.

Approach: You are allowed to use any classifier algorithms covered in the class. SVM, Logistic Regression, and Student-t classifier are some possible alternatives. Implementing extensions of these models for multiple instance may help improve their prediction accuracy on this problem. You are allowed to use any normalization and dimensionality reduction techniques.

Submission: You are going to submit a CSV spreadsheet with predicted labels. The order of predictions should match the order of the samples in the *test_sampleid* variable in the *whole.mat* file. You can include up to three columns in this spreadsheet, one column for each of the different approaches you have tried. You will also send a one-page PDF document explaining each approach you used in detail. This document will not be graded but if

Variable name	Size	Description	Type
train_classid	7849x1	Species names as class labels	String
train_feats	7849x384	DiNo Features	Float
train_imgid	7849x1	Image IDs	Integer
train_sampleid	7849x1	Specimen IDs	Integer
val_classid	1379x1	Species names as class labels	String
val_feats	1379x384	DiNo Features	Float
val_imgid	1379x1	Image IDs	Integer
val_sampleid	1379x1	Specimen IDs	Integer
test_feats	2454x384	DiNo Features	Float
test_imgid	2454x1	Image IDs	Integer
test_sampleid	2454x1	Specimen IDs	Integer

Table 1: Variables in the $whole.mat$ file.

Variable name	Size	Description	Type
train_classid	70641x1	Species names as class labels	String
train_feats	70641x384	DiNo Features	Float
train_imgid	70641x1	Image IDs	Integer
train_sampleid	70641x1	Specimen IDs	Integer
train_tileid	70641x2	2D Tile IDs	Integer
val_classid	12411x1	Species names as class labels	String
val_feats	12411x384	DiNo Features	Float
val_imgid	12411x1	Image IDs	Integer
val_sampleid	12411x1	Specimen IDs	Integer
val_tileid	12411x2	2D IDs for each image part	integer
test_feats	22086x384	DiNo Features	Float
test_imgid	22086x1	Image IDs	Integer
test_sampleid	22086x1	Specimen IDs	Integer
test_tileid	22086x2	2D IDs for each image part	Integer

Table 2: Variables in the $parts.mat$ file.

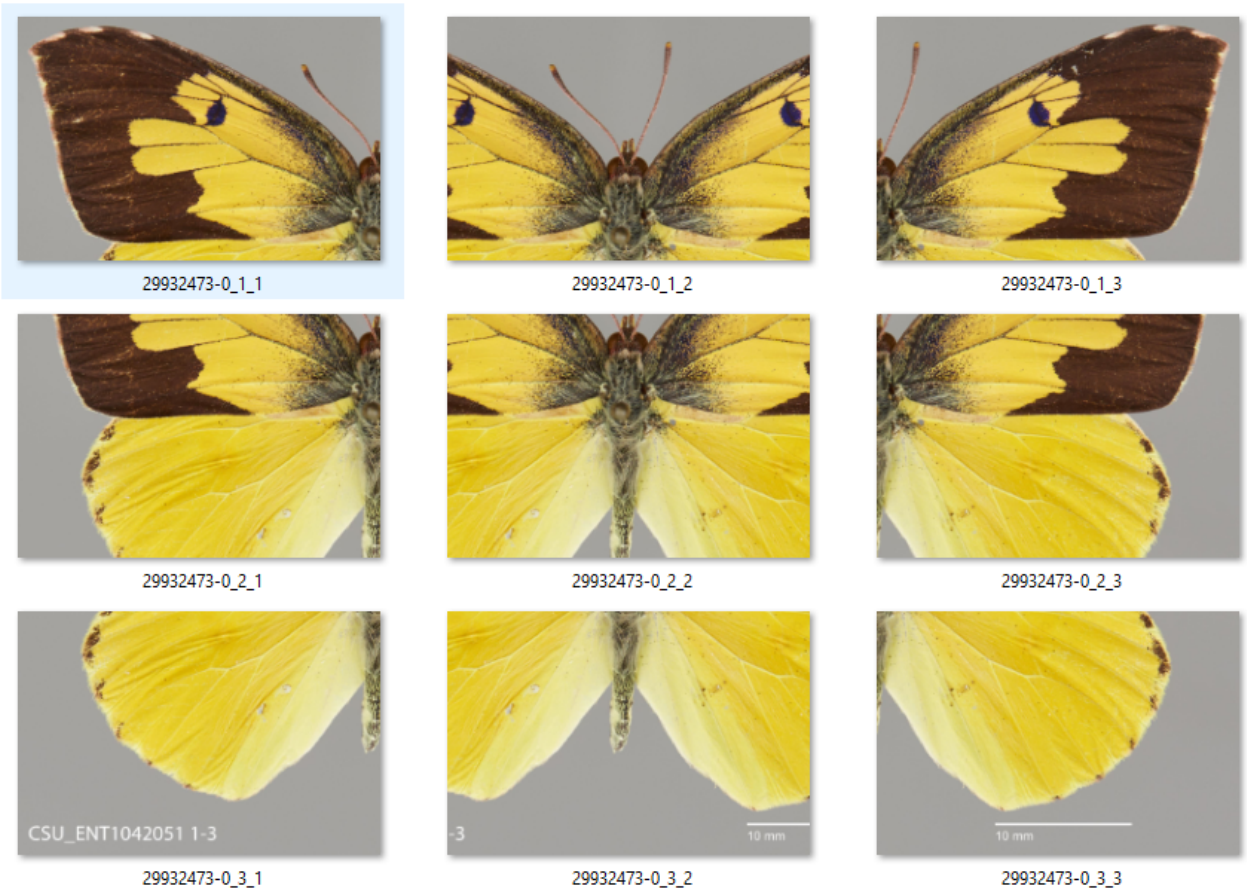

Figure 1: Image is split into nine parts by a 3×3 grid with 50% overlap between parts. Part IDs are the last two numbers in the image file name in the figure.

not submitted may disqualify your submission. Your submissions should be reproducible. Your training and testing scripts may be requested if needed.

Evaluation: This is the same as Task 1. Your submission will be evaluated by mean class accuracy. For each class in the testing data true positives (TP) and false negatives (FN) will be obtained and class-level accuracy will be computed as $\frac{TP}{TP+FN}$. Mean class accuracy will be used for final evaluation and ranking. Top performer will receive the top score of 75 points. The rest of the submissions will be scored relative to the top performer.