(/)

LOGOUT

OPEN HACK ENVIRONMENT OVERVIEW OPEN HACK GUIDE

PROVIDE FEEDBACK

1 2 3 4 5 6

Mark Complete

Challenge 4: Following the Marked Trail

Background

Previously, you build a convolutional neural network (CNN) to classify product images. A CNN typically consists of multiple *convolutional*, *pooling*, and *drop* layers that extract features from images, and one or more dense *fully-connected* layers that map those features to classes.

Transfer Learning is a commonly used machine learning technique in which you can leverage the feature extraction layers from an existing model, and add your own fully-connected layer to predict classes from the extracted features.

Prerequisites

- A Data Science Virtual Machine (DSVM)
- The resized ***gear*** image data fom the previous challenges.
- An installation of the latest version of your chosen deep learning framework(s) based on the **References** section below.

(/)

Challenge

LOGOUT

There are two elements to this challenge:

1. Use transfer learning to train a classifier based on an existing model.
2. Use your model with new data.

1. Use transfer learning to train a model

Create a new CNN by using *transfer learning* to build a classifier on top of the feature extraction layers defined in an existing model.

Hints

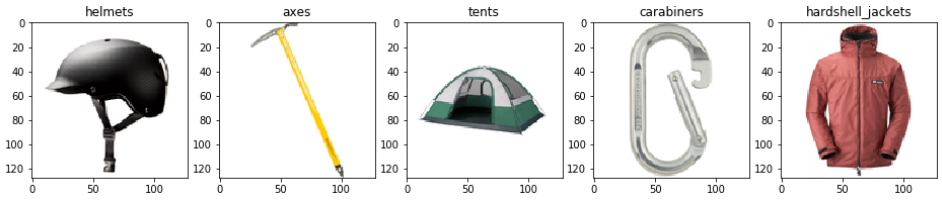
- You can use any base model supported by your chosen deep learning framework.
- You should "freeze" the feature extraction layers in the base model to use their existing trained weights - you need only train the custom layers you add for classification.
- You may need to resize the images to match the size used to train the base model you select.

2. Use your model with new data

Use your model to predict the class of at least five images that are not included in the ***gear*** dataset. You can use the same five images you found in the previous challenge.

Success Criteria

- Successfully train a CNN based on an existing trained model.
- Achieve model accuracy of **0.9** (90%) or greater using your test data set.
- Show predictions for the five images you identified in the **Challenge** section, like this:



Gear predictions

(/)

(Note: Your model is not required to predict the correct class for all of the images, but it would be good if it does!)

LOGOUT

References

- Transfer Learning Notes (http://cs231n.github.io/transfer-learning/)
- Transfer Learning with PyTorch (https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html)
- Transfer Learning with Keras (https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html)
- Transfer Learning with TensorFlow (https://www.tensorflow.org/hub/tutorials/image_retraining)
- Transfer Learning with CNTK (https://cntk.ai/pythondocs/CNTK_301_Image_Recognition_with_Deep_Transfer_Learning.ht