PowerAI Vision Workshop Exercise 1 - Image classification

In this exercise you will practice Image Classification. With Image Classification, a ML algorithm learns to assign categories (classes) to images. The goal of the exercise is to familiarize with the user interface, Image Classification related tasks and the overall process. We will use a Dogs dataset

1. Creata a **dataset** called CvB (for Chihuahuas vs Boxers)
    a. On the welcome screen, press 'Get Started' (alternatively, select Data Sets from the menu bar). The Data Sets screen appears.
    b. Press the plus sign to 'Create a new dataset' on the leftmost tile. (.zip file upload is valid only for datasets exported from PAIV). Name the data set 'CvB'.
    c. Click on the tile 'CvB'. A 'Data set / CvB' screen will open.
    d. Click on 'Import files' under the tile 'Drop files here'. (Use 'Import' for opening the file browser; you can select .zipped collections, or separately browsed images)
    e. Select the file *-boxer.zip and upload it. The status message will inform you about the success.
    f. At the bottom of the pane, select 'Items per page: 100' to view as many images as possible. This will improve productivity.
2. Create and assign **categories (classes)**
    a. Select all pictures by clicking the checkbox right above the images.
    b. Select 'Assign category' and specify 'Boxer' as the class. A new category will be created and selected images will be labeled with the category name
    c. On the leftmost pane, Click on Categories and select 'Uncategorized'. The right pane will become empty.
    d. Upload images of the *-chihuahua.zip file usinge the previous method.
    e. After successfully uploading them, select all images (do not forget to specify 'uncategorized' – unlabeled images will have no category assigned, and this helps you easily assign class names with as few clicks as possible)

f. Unselect some of the images, and assign the new category ’Chihuahua’ to the selected ones.
g. Make another fictional category (Chiwa) and assign the remaining pictures to this category one by one.

3. **Augment** data
    a. Select the ’chihuahua’ images
    b. Select All
    c. Click ’Augment Data’
    d. Select Noise, Horizontal Flip with the parameters unchanged. Select Rotation with 10 degrees and Cropping with 80%.
    e. Observe at the bottom of the window the new dataset size. Flipping creates one additional picture, other filters create 5 additional images by applying the filter with 5 different random settings.
    f. Click Continue.
    g. Name the new dataset ’CvB_augmented’.
    h. Once augmentation is complete, a new data set is created. This new dataset contains the original images **plus** the augmented ones. Successful augmentation preserves the image set information and increases the data set size, at the same time introduces enough variability to decrease high variance errors.
4. **Train** your model
    a. In your dataset, press the ’Train’ button.
    b. Name the new model ’CvB classifier’.
    c. Select Image Classification as the Type of Training.
    d. Set ’max iteration’ to a value between 600 and 800
    e. Leave other parameters unchanged.
        i. Note: Advanced parameters can improve but even inhibit successful training. Base models are pretrained models that can speed up the training process if there are similarities between the actual data set and the base model.

f. Press ’Train’ to schedule training, and monitor the training process. After initialization the training starts and training loss is shown vs iteration.Training loss and accuracy improves, whereas Test Loss varies.
g. Once Train Loss is eliminated, your model learned what can be learned from this data. Test Loss may not be very good. This can be a sign of overfitting (insufficient data). Wait until the model training completes.

5. **Evaluate** the model.
    a. Click ’Model details’
    b. Check the training results (evaluation metrics) and the confusion matrix (it is only visible if the Advanced metric option is set to ’On’).
    c. Observe the Categories and the included images. You may see that one category is overrepresented (for example due to augmentation). In such cases there is a chance that the classifier becomes skewed. (Understands the category much better than the rest). It would be a good idea to equally augment each category.
        i. Note: Exported models can be saved or transferred to another PAIV machine (e.g. for inference). The model is created as a password-protected .zip file. The password is undisclosed, and the .zip file cannot be used elsewhere.
        ii. Note: Custom models based on Tensorflow can be exported. They can be deployed anywhere, not only on PAIV systems.
6. **Export** the model
    a. Under Models, select the one you just created and click ’Export model’. It will be saved on your machine.
7. **Deploy** and **test** your classifier.
    a. Click ’Deploy model’. Assign a name to the deployed model.
    b. After the deployment finishes, click the name of your newly created model.
    c. Observe that an API endpoint is created that accepts your service request and ready to use the classifier for data sent.
    d. Upload an image from the input data.
    e. The model assigns a category, a confidence level and shows on the heat map which part of the image contributed most to the final decision.

i. Note: this training, although accuracy could have reached a high value, may not be an excellent predictor. It produces errors and does not generalise well.

f. Test with different images.

8. **Import** the model
    a. From the Deployed models menu option, select the deployed model and delete it. Under models, select the model and delete it.
    b. Under Models, select Import zip file, and specify the previously saved model. Deploy and test it. It should behave exactly the same as previously.
    c. Try to transfer models within the class.
9. **Exporting** and **importing** the datasets.
    a. From the labeled dataset (click on the tile under Data Sets) select export and save the exported dataset on your desktop.
    b. Extract the zipped file – do not delete the zipped version.
    c. Examine the zip content, the filenames and peek into the 'prop.json' file.
    d. Delete the dataset on PAIV.
    e. Import the labelled dataset from the .zipped file. Check that all labels are restored.
        i. Note: Importing external labelled datasets has some limitations. Refer to the documentation page at:

https://www.ibm.com/support/knowledgecenter/en/SSRU69_1.1.3/base/vision_create_dataset.html

This concludes the exercise.