**COS 529 Advanced CV Assignment 2**

Kunj Mehta (km6838)

**Brief Problem Statement**

In this assignment, we are supposed to build a zero-shot recognition system to classify animals using the Animals with Attributes dataset. In short, this dataset contains 37332 images of 50 different animal classes, each being described with 85 different attributes. The train-test split for the data is ~31K training images and ~6K images as test images.

These are the files we are given:

- `trainclasses.txt:` list of the training classes
- `testclasses.txt:` list of the test classes
- `classes.txt:` list of all the classes
- `predicate-matrix-binary.txt:` binary matrix associating presence of attributes to each animal class
- `test_images.txt:` test images

Here, the file `predicate-matrix-binary.txt` is very important as that is what we will be using to map the image features that are learnt to classes. Meaning the attributes will act as an intermediary and help the model attain zero-shot ability.

**Description of Approach**

Now, because we are supposed to build a zero-shot recognition model, we need to teach the model not only what each image is made of (the image features) but also to predict or recognize those features themselves. This is where the attributes / metadata come in.

We make the model learn how the image features that it extracts map to the 85 attributes, and based on the attributes that are predicted by the model, we output the corresponding classes. In this way, when we want to add another animal class, we can just add the required attributes for that class in the `predicate-matrix-binary.txt` file.

*Detailed Approach:*

- **Data Loader**: The data loader that I work with loads the train / test images, their corresponding target attributes using `predicate-matrix-binary.txt` , the class ID and the class name. The class ID and the class name serve as the target for the whole end-to-end pipeline of zero-shot recognition while the attributes serve as the target for the feature extractor + classifier (ResNet-50) model.
- **Feature extractor**: vanilla ResNet-50 *(not pretrained)*. I use the features extracted by ResNet-50 just before the fully connected layer. I train this model on the training set for 20 epochs.
- **Classifier:** For classification, a Sequential layer of BatchNorm, Dropout of 0.25 and a FC layer mapping feature embedding dimension to a 85-shape tensor representing

probability of the attributes being 1 is used. The attribute logits that are predicted are squished to a probability by using a Sigmoid function. In essence, we are predicting the presence of 85 individual and independent values. (Multi-label classification)

- **Loss**: I use the Binary Cross Entropy loss because as mentioned above, the model is used to predict the values of the class attributes. Since the class attributes are binary (0 or 1), I use BCE loss, which is calculated on the target 85 attribute *flags* and the predicted 85 attribute *logits*
- **Zero-shot recognition**: To obtain the animal class from the predicted attributes, I experiment with metric learning. I use Euclidean distance and Cosine distance scores to find out which one of the target attribute sequences (for all 50 classes) is closest to the predicted, and output the class corresponding to this closest attribute sequence.

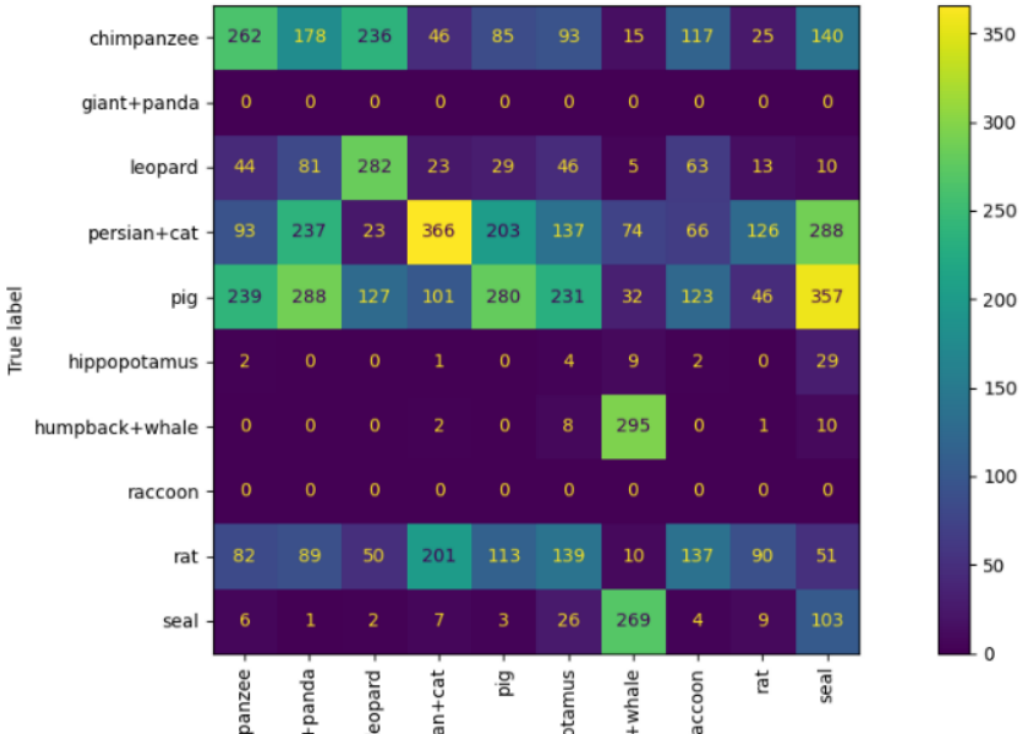
**Accuracy**

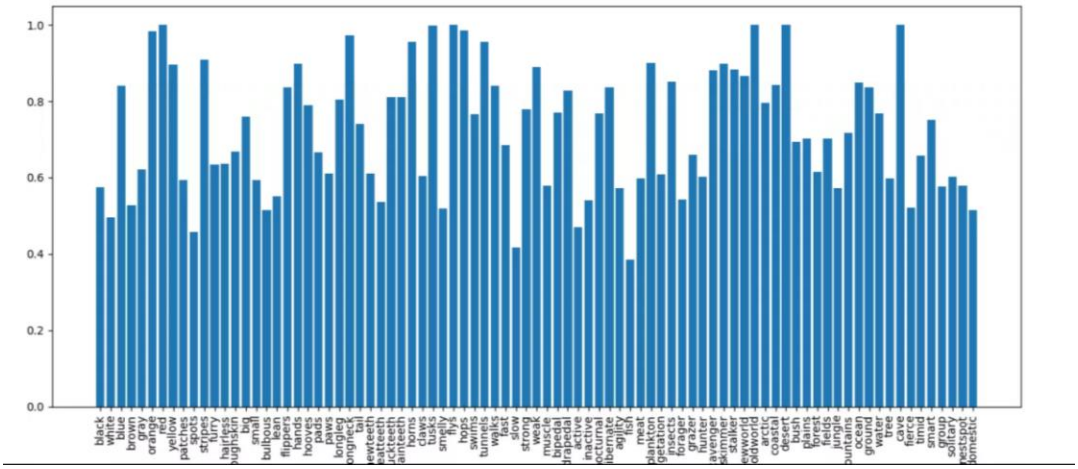
As mentioned above, I tested two distance metrics with the best performing epoch of the ResNet-50 feature extractor and obtained these accuracies:

- **Cosine Distance:** 0.02 (2%)
- **Euclidean Distance:** 0.241 (24.1%)

**Error Analysis**

The below results are for ResNet-50 with Euclidean distance as metric for zero-shot recognition






We can see from the above plots that the ResNet-50 model confuses:

- chimpanzees with pandas and leopard
- persian cat with pig and pandas
- Pig with seal
- Seal with humpback whale

We also see that the model is not able to classify raccoons and pandas while it does extremely well in classifying humpback whales.

From the second plot, we can see the accuracy of the feature extractor in predicting individual image attributes (a logit threshold of 0.5; anything above 0.5 is 1). We can see that the feature extractor is not able to recognize whether

- An animal is ‘black’
- An animal is ‘white’
- Is a ‘fish’
- Animal is ‘slow’
- Has ‘spots’
- Is ‘active’

The biggest sources of error in my zero-shot recognition system are for the classes ‘pig’ (which is often confused with ‘seal’, and everything else), rat (again confused with everything else in equal measure) and ‘seal’.

The model is also not able to classify ‘pandas’ and ‘raccoons’. Part of this is because of the class imbalance inherent in the dataset and the way the embeddings seem to be clustered [2]. ‘Rat’ and ‘pig’ have significantly low number of samples in the dataset in comparison to others.

**Next Steps**

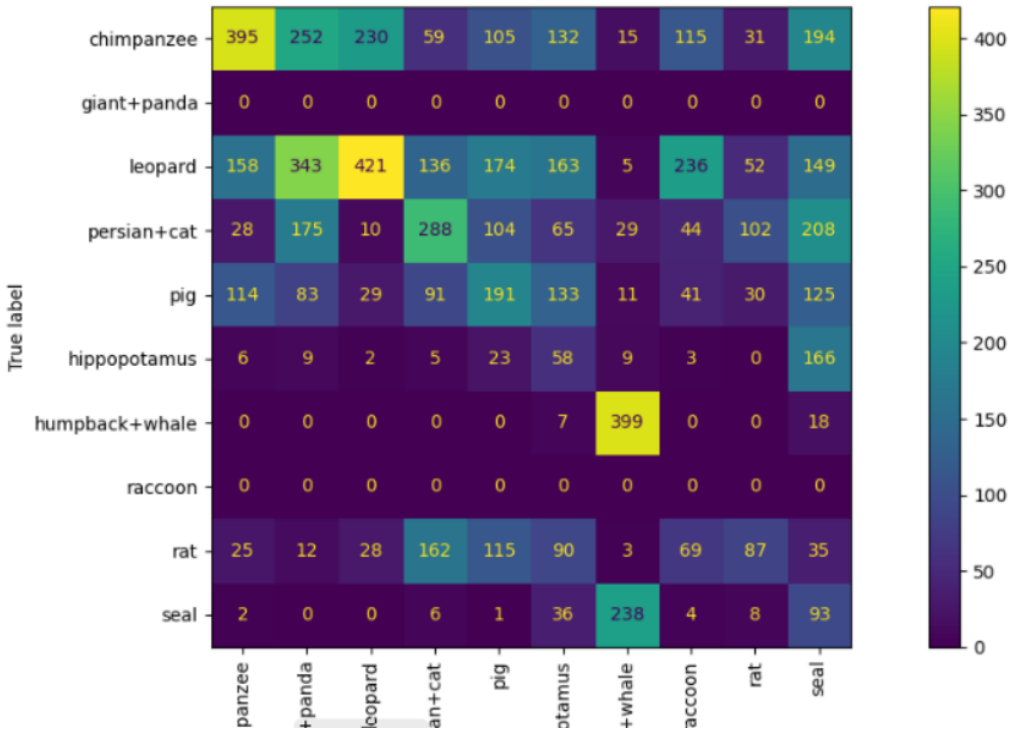
Some next steps I would try are:

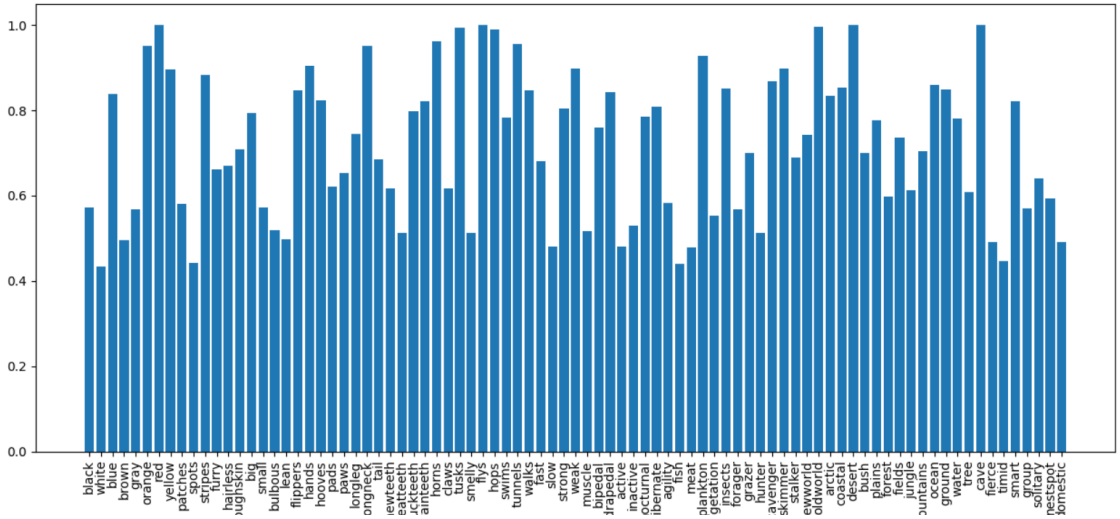
- Try to handle the class imbalance in the dataset in an aim to build a better feature extractor and attribute classifier by weight balancing the training dataset in proportion to the number of examples for each class.
- Experiment with better inherent feature extractors like ResNext-101 and RepVGG [3]. This is because RepVGG is 83% faster in inference and 1 percentage point better in top-1 accuracy than ResNet-50 on ImageNet. ResNext-101 is 2 percentage points better than ResNet-50 in top-1 accuracy on the ImageNet dataset.

  In fact, I experimented by replacing ResNet-50 with RepVGG, trained for 20 epochs with identical hyperparameters and I obtained an accuracy of **0.277 (27.7%)** when using Euclidean distance.

  Overall, in both ResNet-50 and RepVGG, usingEuclidean distance gives better performance than cosine distance.

  The confusion matrix and attribute accuracy plot for RepVGG is below.






Comparing the two confusion matrices for both models, we can see accuracies for ‘humpback whale’, ‘leopard’, ‘chimpanzee’ all rise while confusion that was apparent in classes like ‘pig’, ‘persian cat’ and ‘rat’ decreases.

**References**

[1] C. H. Lampert, H. Nickisch, and S. Harmeling. "Learning To Detect Unseen Object Classes by Between-Class Attribute Transfer". In CVPR, 2009

[2] Y. Xian, C. H. Lampert, B. Schiele, Z. Akata. "Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly", T-PAMI 2018

[3] Ding, Xiaohan and Zhang, Xiangyu and Ma, Ningning and Han, Jungong and Ding, Guiguang and Sun, Jian. “Repvgg: Making vgg-style convnets great again”, CVF 2021.