Zero-shot Learning Using Direct Attribute Prediction Method

DL for Spiking Neural Networks and Advanced Data Mining
Final Project Report

Based on the paper "Unseen Object Classes by Between-Class Attribute Transfer"
by C.H. Lampert, H. Nickisch, and S. Harmeling. [1]

Author
Weronika Zawadzka 12244068

01.07.2023, Klagenfurt University
Summer Semester 2023

Introduction

The field of object recognition has developed largely over last decades. Training such systems requires high number of well-labelled training examples. However, data extraction (generation) and labelling are both tedious as well as financially and time costly procedures. Moreover, in the vast regions of interest for such models, there could be indefinite number of areas with an unspecified or lacking data to work on. With help comes ZSL, zero-shot learning technique. "The core in ZSL is to learn a multi-modal projection between visual and semantic features, by using labeled seen class data only. Then, the projection is able to generalize well to handle unseen class data in the test stage." [2] That it to say, it is possible to make predictions about classes not present during the training phase. Main idea is presented on Figure 1 below.

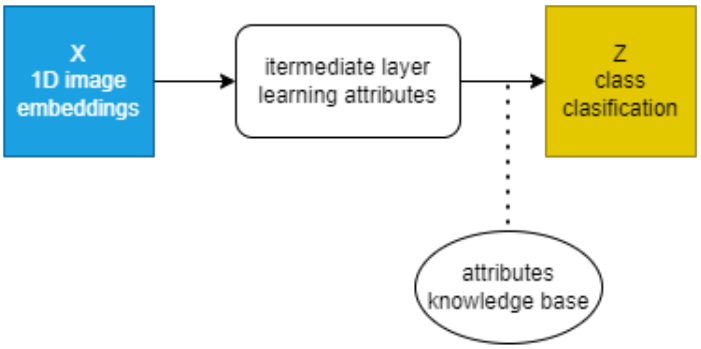

Figure 1: High level overview of DAP ZSL solution

Datasets description

For this project two datasets will be used, namely AWA2 [3] and CUB-200-2011 [4]. AWA2, Animals with Attributes 2, is a dataset containing 37322 images of 50 animals with their attributes. For each animal, there are 85 attributes like color, stripe or habitat. Split proposed by authors will be preserved. The CUB dataset, namely Caltech-UCSD Birds-200-2011, is a bigger dataset containing 11,788 images of 200 subcategories belonging to birds, each having 312 different attributes. For this dataset, there is also preferred train/test split, and so it will be kept throughout the experiments. As inputs, instead of images themselves, lower-dimensional (1D)

representation of the image will be used as described in AWA2 paper: "Our image embeddings are 2048-dim top-layer pooling units of the 101-layered ResNet".[3] The labels are integer values representing classes. When it comes to attributes, for AWA2 binary attributes are available, and so 1 means presence of a given attribute of set of 85, 0 - absence. For both AWA2 and CUB continuous representation of per-class attributes (between 0 and 1) is also available, which showed [5] to be stronger than binary ones. Training and testing classes are disjoint so that testing classes have not been seen in the training part - see Figure 1 and 2.

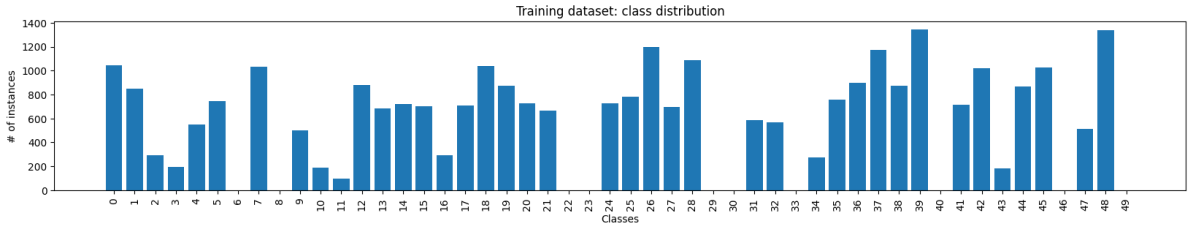

Figure 2: Classes distribution of AWA2 train dataset

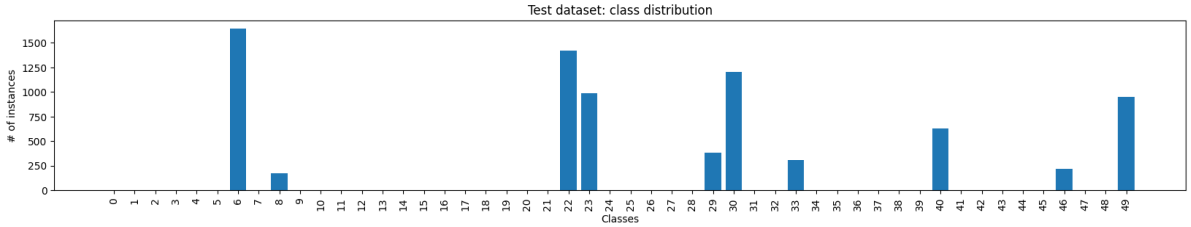

Figure 3: Classes distribution of AWA2 test dataset

Plan

Goal: Classify test images into classes which have never been seen
Method: Learn attribute classifiers (or regressor) for image representations and then use a mapping from predicted attributes to class

The idea is to use both binary and continuous representations of attributes and compare the results between them. The chosen method is DAP - Direct Attribute Prediction [1] which is traditional attribute-based prediction algorithm. It is an intermediate method, as it requires an classifier (or a regressor) from images to attributes and only at test time the classes names can be inferred, so that decision is made based solely on the attribute layer. For CUB dataset, only continuous attributes are available, so only they will be used. However, for AWA2 both models involving classification for binary attributes and regression for continuous one will be build. Supervised classifiers/regressors can be created using different methods, e.g. SVM/SVR or Neural Nets. Neural Nets approach will be used in this project.

Problem formulation

$$\text{Let } X = \text{images } \textit{embeddings}$$

$$Y = \text{classes seen at } \textit{training}$$

$$Z = \text{classes unseen at } \textit{training}$$

The task is to learn a classifier

$$f : \mathrm{X} \rightarrow \mathrm{Z}$$

by using of training examples and intermediate layer of attributes.

$$\text{Let } A = \text{images } \textit{attributes}$$

$$\text{And } \alpha \in \text{A - an attribute representation}$$

Then, in the intermediate layer a classificator (or regressor) from image embeddings to attributes is created. From predicted attributes classes can be inferred:

$$\alpha : \text{X} \rightarrow \text{Z}$$

Part 1: Classification

For the classification in the intermediate layer binary attribute representations are used. A neural model with 85 (for AWA2) singular probabilistic classifiers for each of the attributes is build. Sigmoid (1) is used as the activation for each of the attributes classifiers.

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{1}$$

In the paper, for the mapping from predicted attributes from the model to unseen classes MAP prediction is used, that assigns the best (highest score) output class from all test classes to a test sample. However, it is said that setting the priors of attributes to be uniform yields comparable results. Therefore following decision rule has been applied:

$$\text{Let } m_1, m_2, m3, ..., m85 \in \alpha$$

So that each one represents given singular attribute out of 85 avaiable in AWA2 dataset. Also,

$$\textit{Let } \alpha^p \textit{ be a predicted attribute representation}$$

Then:

$$p(a_m^p = a_m^z | x^z) = \begin{cases} a_m^p, & \text{if } a_m^z = 1 \\ 1 - a_m^p, & \text{otherwise} \end{cases} \tag{2}$$

And so the class prediction z* can be done by:

$$z^* = argmax_z \prod_m p(a_m^z | x^z) \tag{3}$$

Method

5 different classification models has been build and the one with the highest accuracy score has been chosen (see Listing 1). The highest accuracy (on AWA2) turned out to be 41.9%, which may seem as a low number, but in the the implemented paper [1] authors got 40.5% on the AWA dataset using DAP method. For the comparison, in the AWA2 paper, the authors reported [3] the score of 46.1% on AWA2 and 44.1% on AWA1 dataset. Parameters of the model focus on prevention of overfitting (L1 regularizer, high dropout rate, lower learning rate) as the images are already pre-trained embeddings from ImageNet.

```
x = Dropout(0.7)(inputs)
attribute_classifiers = []
for p in range(85):
        single_classifier = Dense(1, activation='sigmoid',
                activity_regularizer=l1(0.01))(x)
        attribute_classifiers.append(single_classifier)

model = Model(inputs, attribute_classifiers)
model.compile(optimizer=Adam(learning_rate=0.001), loss=
['binary_crossentropy']*85, metrics=['accuracy'])
model.fit(trainDataX, list(trainDataAttrs.T), batch_size=64,
shuffle=True, epochs=10, callbacks=
[EarlyStopping(monitor='loss',min_delta=0.0001, patience=2)])]
```

Listing 1: Classification model for AWA2 ZSL

Part 2: Regression

For regression, where continuous representations of attributes are used, simple measure of sum of absolute differences has been applied as to the score of similarity between a prediction and each per-class attribute representation:

$$l(a_m^p|x^z, a_m^z) = abs(a_m^p - a_m^z) \tag{4}$$

And so the class prediction z* can be done by:

$$z^* = argmin_z \sum_m l(a_m^z|x^z) \tag{5}$$

Method

Instead of multiple attributes classificators models, a regression one has been used. For the regression done on AWA2 dataset the same methodology was used, that is creating 5 different models and choosing one with the best accuracy (see Listing 2). Instead of sigmoid activation, linear one has been used. The accuracy with continous attributes increased from previous value of 41.9% to 47.3%

.

```
x = Dropout(0.8)(inputs)
x = Dense(85, activation='linear')(x)

model = Model(inputs, x)

with tf.device(device_name):
  model.compile(optimizer=Adam(learning_rate=0.0005), loss='mae', metrics=['mae'])

hist = model.fit(trainDataX, trainDataAttrs,
batch_size=64, shuffle=True, epochs=10, callbacks=[EarlyStopping(monitor='loss',
min_delta=0.0001, patience=2)])
```

Listing 2: Regression model for AWA2 ZSL

For the CUB dataset, another model has been build - see Listing 3. The highest accuracy achieved during parameters experiments is 34.7%. In comparison in [3] the results achieved by authors were of 40%. So, results achieved here turned out to be a bit worse despite testing many variants of the model during experiments process.

```
x = Dense(700)(inputs)
x = Dense(312, activation='linear')(x)

model = Model(inputs, x)

model.compile(optimizer=Adam(learning_rate=0.0005), loss='mae', metrics=['mae'])

hist = model.fit(trainDataX, trainDataAttrs,
batch_size=64, shuffle=True, epochs=10, callbacks
[EarlyStopping(monitor='loss',min_delta=0.0001,
```

Listing 3: Regression model for CUB ZSL

Results

Accuracy results obtained by methods described in previous sections have been summarized in Table 1.

Accuracy scores comparison		
Dataset	Binary attributes	Continuous attributes
AWA2	41.9%	47.3%
CUB	not applicable	34.7%

Table 1: Comparison of ZSL accuracy scores of DAP method

Moreover, as for AWA2 dataset both binary attributes (so classifiers intermediate layer) and continuous attributes (regressor) we can compare them using confusion matrix - see Figure 4 and 5.

Observations

An interesting observation is that for example bat class has not been predicted even once when used classificators method (see Figure 4). It changed in the regressor method, but very slightly. Overall, usually the mistakes of the model made sense, like mistaking blue+whale for seal or blue+whale for dolphin. What has been seen during training is that the attributes classificators would learn at different speeds, as expected because of attribute and class imbalances (Figure 3). For some some attributes even in the learning phase not much examples were present, and so it was hard to learn them and not overfit more frequently seen ones at the same time. This is the disadvantage of learning all 85 different attributes classifiers simultaneously.

Conclusions

Summing up, the continuous representation of attributes does in fact yield better results than its binary counter method. Zero shot learning problem could be solved, however not with the

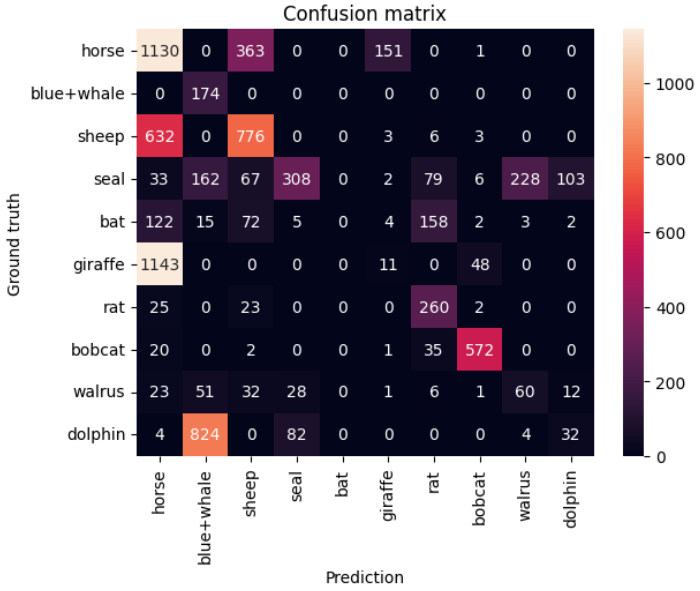

Figure 4: Confusion matrix of DAP method for binary attributes of AWA2 dataset

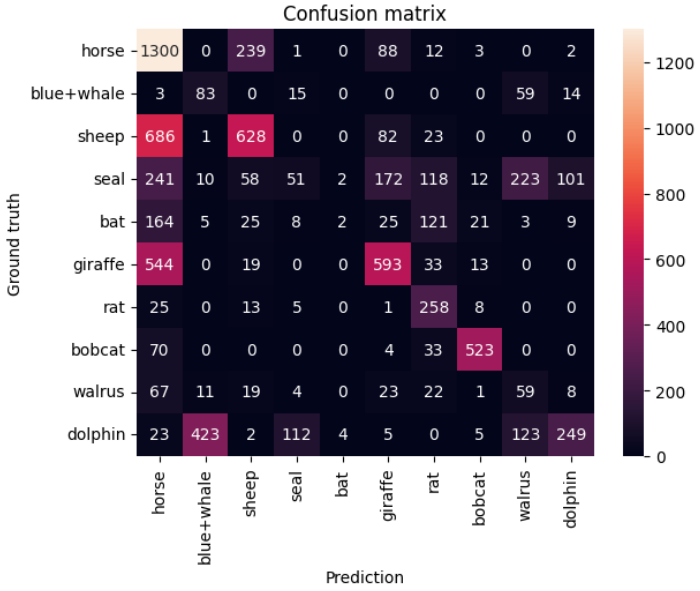

Figure 5: Confusion matrix of DAP method for continuous attributes of AWA2 dataset

highest accuracies score. It is important to note that direct attribute method is the most simple one, used often as a baseline, and therefore highest scores were in fact not expected. Much more advanced methods than DAP were proposed that present better results. Future work could be to compare more methods between each other.

Bibliography

[1] C. H. Lampert, H. Nickisch and S. Harmeling, "**Learning to detect unseen object classes by between-class attribute transfer**," 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 2009, pp. 951-958, doi: 10.1109/CVPR.2009.5206594.

[2] Y. Liu and T. Tuytelaars, "**A Deep Multi-Modal Explanation Model for Zero-Shot Learning**," in IEEE Transactions on Image Processing, vol. 29, pp. 4788-4803, 2020, doi: 10.1109/TIP.2020.2975980.

[3] Xian, Y., Lampert, C. H., Schiele, B., Akata, Z. (2020). **"Zero-Shot Learning – A Comprehensive Evaluation of the Good, the Bad and the Ugly"**. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/1707.00600

[4] Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S. (2011). , Author = Wah, C. and Branson, S. and Welinder, P. and Perona, P. and Belongie, S., **CUB dataset**, Year = 2011 Institution = California Institute of Technology, Number = CNS-TR-2011-001. California Institute of Technology.

[5] Z. Akata, F. Perronnin, Z. Harchaoui and C. Schmid, "**Label-Embedding for Image Classification**," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 7, pp. 1425-1438, 1 July 2016, doi: 10.1109/TPAMI.2015.2487986.