Analysis on Inter/Intra Class Prototypes in the Latent Space

Yash Patel
Dr. Dilip Sarkar
Nathaniel Dean

Department of Computer Science, University of Miami, Coral Gables, FL, USA

[1] Background

Classifiers are models used to classify inputs, which can be in the form of audio, imagery, or text, into a specific category. Traditionally, classifiers in machine learning are evaluated using a test set, which is a set of data similar to the data the classifier was trained on that the classifier had not previously seen. This is particularly the case in classifying Neural Networks, which are models that utilize machine learning and training data for the purpose of accurately classifying future inputs. Classifiers can generate class prototypes[1], which may reflect the features with which their model identifies a specific class. In the case of image classifying neural networks, this can be done by generating one random image corresponding to each class, and updating each image such that the networks classifies it into its corresponding class with 100% confidence. This is done by assigning the target for the image's classification to its corresponding target class, and utilizing cross-entropy loss to update the image's gradients so the network has more confidence that the image classifies into its target class.

[2] Objective

With the goal of further analyzing prototype images and their relationships over different intervals of model training, we create intermediary prototypes, where the intermediary initializes at one prototype, and trains towards another. Then, we extract the "feature vector" of the image as it is passed through the classifier, and observe its dissimilarity[2] to both its origin prototype, its target prototype, and other intermediaries that share its origin or target.

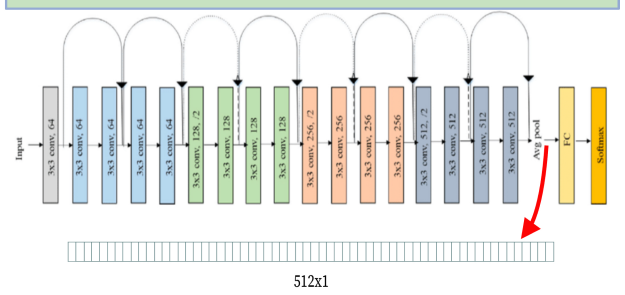


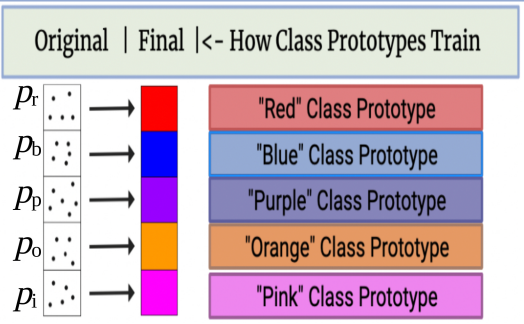
Figure 1: Demonstrating the point in the network at which the feature vector is extracted, prior to classification

[3] Architecture

In these observations, we use the ResNet18[3] deep neural network architecture with the Cifar10 and Cifar100 datasets. Class prototypes are trained in intervals, which are separated by further training of the model. Prototypes are saved at each interval, for the purpose of observing how the inspected metrics may change as the model is further trained and gains accuracy.

[4] Intermediary Prototype Creation

For a classifier that classifies into n classes, there are n class prototype images, denoted as $\{p_1, ..., p_n\}$. For each class prototype, there are $n-1$ intermediary prototypes that may be generated. Each intermediary initialized as p_j, which is trained towards target class k, $\forall\ k \in \{0,...,n\}$ where $j \neq k$, may be represented as $p_{k,j}$. Each intermediary prototype is passed through the model and then updated using Cross-Entropy Loss, with a learning rate of 0.1(for Cifar100) or 0.01(for Cifar10). It is updated until the loss term is below 0.01. All intermediary prototypes classify into their target class after the conclusion of their updates. Afterwards, they are saved into an $n \times n$ matrix P, whose indices can be identified as $P_{k,j}$ where $P_{k,j} = p_{k,j}$. The principle diagonal of P, the set of indices $P_{k,j}$ where $k = j$, is filled with the class prototype p_k.



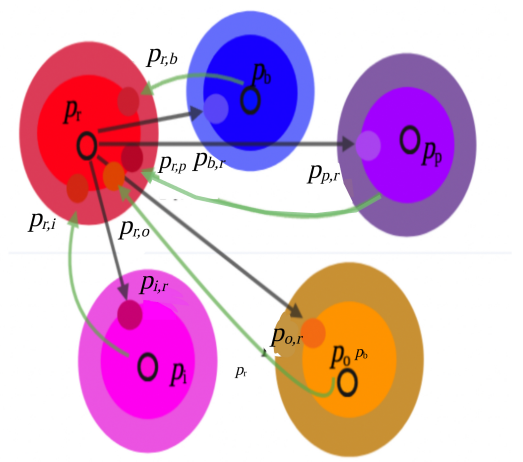


Figure 3: An example of a classification space with 5 classes, demonstrating how intermediary prototypes may appear

[5] Tracked Metrics

The dissimilarity of two feature vectors is measured in these observations by first computing their cosine similarity, and subtracting it from 1. Thus, values may range from 0 to 1, and a higher value denotes greater dissimilarity.

With P, two $n \times n \times n$ matrices of dissimilarities are generated

Matrix C: Inter-Class Prototype Dissimilarities

Inter-Class dissimilarities are taken between intermediary prototypes which were originally instantiated at the same class j, where the dissimilarities between each prototype $P_{k,j}$ and the prototypes $\{P_{0,j},...,P_{n,j}\}$ will be stored in $\{C_{k,j,0},...,C_{k,j,n}\}$.

Matrix *R: Intra-Class Prototype Dissimilarities*

Intra-Class dissimilarities are taken between intermediary prototypes which classify into the same class k, where the dissimilarity between each prototype $P_{k,j}$ and prototypes $\{P_{k,0},...,P_{k,n}\}$ will be stored in $\{R_{k,j,0},...,R_{k,j,n}\}$

[6] Example Matrix *P* with *C* and *R* Sub-Arrays

Where *P* is 5x5 and representative of the example classifier of colors where:

Class 1 = Red
Class 2 = Blue
Class 3 = Purple
Class 4 = Orange
Class 5 = Pink

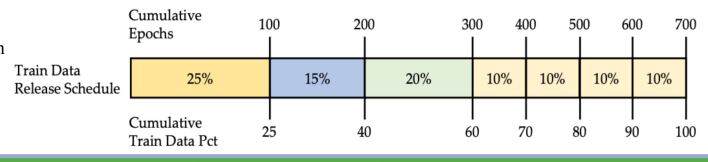
$P =$

$p_{r,r}$	$p_{r,b}$	$p_{r,p}$	$p_{r,o}$	$p_{r,i}$
$p_{b,r}$	$p_{b,b}$	$p_{b,p}$	$p_{b,o}$	$p_{b,i}$
$p_{p,r}$	$p_{p,b}$	$p_{p,p}$	$p_{p,o}$	$p_{p,i}$
$p_{o,r}$	$p_{o,b}$	$p_{o,p}$	$p_{o,o}$	$p_{o,i}$
$p_{i,r}$	$p_{i,b}$	$p_{i,p}$	$p_{i,o}$	$p_{i,i}$

$R_{1,2}$ is the array of dissimilarities between $p_{r,b}$ and $\{p_{r,r},...,p_{r,i}\}$

$C_{1,2}$ is the array of dissimilarities between $p_{r,b}$ and $\{p_{r,r},...,p_{i,r}\}$

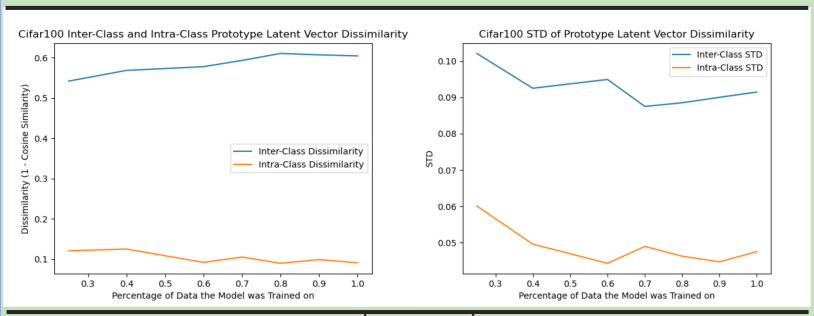
Figure 2: Representation of training intervals used in the study[2]



[7] Results & Conclusions

Over both datasets, we have plotted the average dissimilarities when taken between Inter-Class Prototypes and Intra-Class Prototypes (C vs R), as well as the standard deviations of the dissimilarities. Corresponding to each dataset, there are 7 values of each, derived from class and intermediary prototypes generated at different training intervals of the model. For each dataset, to increase sampling, there are 5 random initializations of class and boundary prototypes, independent of each other. The plotted data are the average of the values observed amongst them.

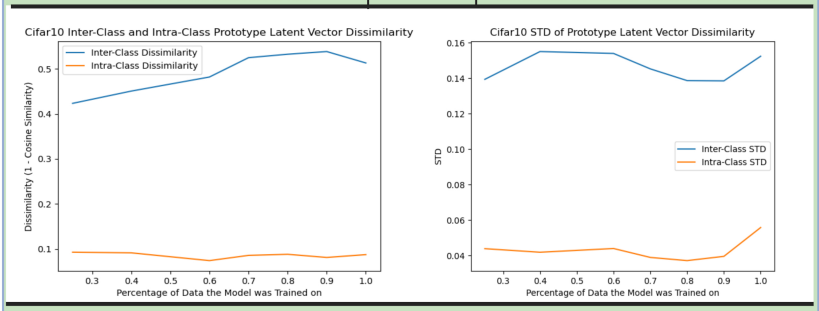
Cifar100



Graphs representing the mean values of C and R over data intervals

Cifar10

Graphs representing the average standard deviation of means values of C and R



It can be seen from both datasets that as the classifier is further trained, the Inter-Class dissimilarity tends to increase, supporting intuition that class boundaries become more distinct as a classifier is more accurate. Intra-Class dissimilarity tends to stay stable; significantly lower than Inter-Class dissimilarity. Furthermore, both standard deviations maintain a similar spread over all training splits, where Intra-Class STD is always lower than Inter-Class STD.

[8] Outliers

We have defined outliers as those dissimilarity values that have a z-score with magnitude greater than 3, with respect to the interval's mean and std.

% Trained	Cifar10	Cifar100
25%	36.92%	7.21744%
40%	41.008%	9.8602%
60%	40.2%	14.38108%
70%	42.36%	8.06404%
80%	40.6%	9.39844%
90%	37.24%	11.1318%
100%	25%	9.48596%

Chart values indicate percentage of data that are outliers. Generally higher in Cifar10, and there is no clear trend shared among the datasets as the model is trained further.

[9] Future Work

Future work aims to further inspect relationships between class prototypes for the purpose of insight into a classifier's robustness. Particularly, identifying those images that lie on the class boundary between every pair of class prototypes. Then, observing their dissimilarity, and how it may vary depending on how much the model has been trained, as well as standard deviation of the dissimilarity. Comparing those values to the values found in this study may provide insight into the shape of the class boundaries that could prove useful in comparing and judging the quality of different classifiers, potentially aiding in the case when one does not have a proper test set.

[10] References & Acknowledgements

1. Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. *CoRR*, abs/1703.05175, 2017.
2. Nathaniel Dean, & Dilip Sarkar. (2023). Fantastic DNN Classifiers and How to Identify them without Data.
3. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

This material is based upon work supported by the National Science Foundation under Grant No. CNS-1949972. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.