# Example Perplexity and Label Cleaning

QIU Yaowen
*The Hong Kong University of Science and Technology*
yqiuau@connect.ust.hk

***Abstract*—Some images are easier to classify than others for human, this should also apply on Deep Neural Networks. Example perplexity is developed to describe the extent of difficulty of classifying examples. Based on the term, an intuitive idea is to modify the training set to make a cleaner training set, in order to improve the generalization ability of models. The project adopted several methods, including percentile threshold pruning methods, mislabeled pruning methods, and weight reassignment methods, to build different versions of training set. However, not the same as expected, none of methods show superiority on the validation set. Even all threshold pruning methods have negative impact on the model, especially on classifying examples with large X-perplexity. Grad-CAM gives a visualization explanation that models trained on pruned dataset fail to focus on right segments of ground truth object. A class-based analysis shows that all models have similar performance on predicting easy and mid-level examples and percentile pruning models have worse performance on predicting hard example. However, based on example analysis, all models tend to give lower entropy and higher probability when classifying easy and mid-level examples.**

***Keywords—Example Perplexity, ImageNet, Data Pruning***

## I. INTRODUCTION

For Deep Neural Networks (DNNs), some images are easier to classify than others. The term perplexity is used to describe the extent of difficulty of classifying. In previous work [1], C-perplexity and X-perplexity are proposed to measure the image perplexity quantitatively. In brief, for a specific example $\mathbf{x}$, C-perplexity means the average uncertainty of the population of classifiers when classifying $\mathbf{x}$, X-perplexity means the fraction of the classifiers that misclassifies $\mathbf{x}$. For a class, the C/X-perplexity is defined to be the average of the C/X-perplexity over all the examples belong to the class. The paper believes that attention confusion and class confusion are two main factors that make an example difficult to classify; visual similarity and class co-occurrence are another two factors that make class perplexity high.

An intuitive idea is to control the level of perplexity of training examples, for example, filter those examples with perplexity larger than a certain threshold. Training examples with relatively lower complexity should lead to better classification result. Similar to this idea, assign examples according to its X-perplexity or C-perplexity may have better classification result on the validation set.

The goal of the project is to adopt several methods based on these ideas to modify the training set and train a model based on modified training set. The project expected that models trained on modified training set have better performance on generalization, compared to baseline model. To verify the effective, models were evaluated on the ILSVRC 2012 validation set and Imagenetv2 dataset.

## II. RELATED WORK

The term perplexity is used to describe the extent of difficulty of classifying. [1] illustrates the concept of example perplexity in details. The dataset used is ILSVRC 2012 [2], which contains about 1.28M training images with 1,000 categories and 50,000 validation images. It is one of the most popular and important visual data set in deep learning and computer vision field. Motivated by the observation that an image is difficult to classify for humans if many people find it confusing and classify it correctly, the paper created a population of $N$ classifiers. First, it re-sampled the training images using proportion of 25%, 50%, and 75%, respectively. With the original training images set, totally 10 different training sets are found. Then, it selected 10 main-stream CNN architectures (i.e., VGG, ResNet, DenseNet, etc) and trained them on aforementioned training image set and save each model in different stages. Totally $N = 500$ models are created.

For a given example $\mathbf{x}$, $P_i(y \mid \mathbf{x})$ is the probability distribution computed by model $i$. The information entropy $H(P_i(y \mid \mathbf{x})) = -\sum_y P_i(y \mid \mathbf{x}) \log_2 P_i(y \mid \mathbf{x})$ is a measure of the degree of confidence of model $i$ classifying example $\mathbf{x}$. The perplexity of the probability distribution is defined to be $2^{H(P_i(y|\mathbf{x}))}$. The C-Perplexity is defined as the geometric mean of perplexity of the probability distribution over $N$ models, as (1) demonstrates. The higher the C-perplexity, the higher the uncertainty or the lower the confidence that a classifier tries to classify example $\mathbf{x}$.

$$\Phi_C(\mathbf{x}) = \left[\prod_{i=1}^{N} 2^{H\left(P_i(y|\mathbf{x})\right)}\right]^{\frac{1}{N}} \tag{1}$$

The definition of X-Perplexity is the fraction of misclassification over $N$ models, as shown in (2).

$$\Phi_X(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(C_i(\mathbf{x}) \neq y) \tag{2}$$

The paper conducted several experiments to show different reasons that make an example has high perplexity. Attention confusion and class confusion are two main factors. Attention confusion refers to there are multiple objects in one example while the ground truth label is only one. Some classifiers focus on one object while another may pay attention on others. If a classifier fails to focus to the object that ground truth label refers to, a wrong prediction happens. Class confusion means objects belong to several classes are similar to each other. For example, Dhole is very similar to wild boar, brown bear, and red fox visually. Even though a classifier successfully locates the ground truth object, it still may make a prediction as other classes. Furthermore, the paper found some examples which are probably mislabeled using X-perplexity and C-perplexity. When X-perplexity is high (says >0.8), but C-perplexity is low (says close to 1), which means all the base classifiers have strong confidence on the prediction but most of them still make wrong prediction. According to this principle, the paper did find some examples which are definitely mislabeled. For example, an example that all classifiers predict as lion are panda and visually looks like a lion, however, it is labeled as hussar monkey, as Fig. 1 demonstrates.

| | (a) | (b) | (c) | (d) |
|---|---|---|---|---|
| ImageNet Label: | lesser panda | ant | hussar monkey | amphibian |
| X-Perplexity: | 1 | 1 | 1 | 1 |
| C-Perplexity: | 1.00 | 1.14 | 1.00 | 1.15 |
| TVL: | giant panda | bee | lion | tank |

Fig. 1. Example of probably mislabeled examples

An intuitive idea is to optimize the training image set using the information of perplexity.

## III. Methodology

### A. Baseline

A baseline model should be built to compare with methods used. Create a baseline model is easy. A Mobilenetv2 [4] model is used to train on the full ILSVRC 2012 training images.

TABLE I. ACCRAUCY AND SPEED COMPARE

| Model | Size (MB) | Top-1 Accuracy | Time (ms) per inference step (GPU) |
|---|---|---|---|
| VGG16 | 528 | 0.713 | 4.16 |
| ResNet50 | 171 | 0.749 | 4.55 |
| InceptionV3 | 92 | 0.779 | 6.86 |
| Xception | 88 | **0.790** | 8.06 |
| MobileNetV2 | **14** | 0.713 | **3.83** |

TABLE I demonstrates a simple compare between MobileNetV2 and some main-stream model. The MobileNetV2 model has the same Top-1 accuracy with VGG16 model while has huge advantage on size and speed. It allows more experiments in limited time.

### B. Percentile Pruning Methods

The idea of pruning is to remove examples that have higher perplexity in the training image set to make a cleaner training set for model. It is similar to remove outliers in the data. In details, for any example $\mathbf{x}$, if either the X-perplexity or C-perplexity is larger than a threshold, $\mathbf{x}$ will be removed.

In the project, the threshold is defined as the percentile of X-perplexity and C-perplexity. Table II demonstrate the statistics of pruned examples under each threshold. When the threshold is set to .95 percentile, 8.36% of examples are removed from the training set. When the threshold is set to .90 percentile, 15.17% of examples are removed. When the threshold is set to .85 percentile, 21.09% of examples are pruned.

TABLE II. STATISTICS OF PRUNED EXAMPLES

| Threshold | # of examples pruned by C-Perplexity | # of examples pruned by X-Perplexity | # of examples pruned (OR) | % of examples pruned |
|---|---|---|---|---|
| .95 | 64,059 | 63,278 | 107,107 | 8.36% |
| .90 | 128,117 | 127,572 | 194,140 | 15.17% |
| .85 | 192,175 | 191,792 | 269,967 | 21.09% |

### C. Mislabeled Pruning Methods

As Fig. 1 shows, mislabeled examples do exist in the training image set. If an example $\mathbf{x}$ belong to $y$ is labeled as $c$, where $c \neq y$, the model learns wrong gradients from the example. The removal of examples that have high probability to be mislabeled may improve the purity of training data, lead to better result.

An example with high X-perplexity and low C-perplexity has higher change to be mislabeled. Consider the distribution of perplexity, the threshold of the X-perplexity is set to 0.8 and C-perplexity is set to 1.2. Any example with X-perplexity larger than or equal to 0.8 and C-perplexity smaller than or equal to 1.2 is removed from the training set. In this pruning method, 997 examples (around 0.08% of training set) is pruned.

### D. X-perplexity based Weight Reassignment Methods

Another idea to optimize the training set is to set different weights for examples according to its perplexity. Mislabeled example is one of reasons that makes example perplexity high. In addition, examples with multiple objects may force the classifier to transfer part of attention on objects that are not ground truth, which may reduce the fitting ability of model. In such as, assign weights that are inversed to the perplexity may increase model's performance. First, the weight reassignment method based on X-perplexity is used.

To assign weight for examples precisely, a function takes the input of X-perplexity and output weights is preferred. Since we want examples with higher perplexity has lower weights, the designed weight function should satisfy two properties. The weight of examples should be monotonically decreasing with its X-perplexity. And the derivative of the weight function should also decrease monotonically. The reason is the classification difficulty gap between an example with X-perplexity 0.8 and an example with X-perplexity 0.4 should be larger than the classification difficulty gap between an example with X-perplexity 0.4 and an example with X-perplexity 0.2.

$$f(x) = b - \frac{c}{e^x} \tag{3}$$

(3) demonstrates the weight reassignment function of examples with two parameters based on given X-perplexity. $b$ and $c$ restrict the minimum value and maximum value of weight. Furthermore, we do not hope the model totally ignore the examples with high X-perplexity. A simple way is to restrict the quotient of maximum weight (i.e., $b - c$) and minimum weight (i.e., $b - \frac{c}{e}$ ) smaller than a number. In the experiment, the gap is default to 2, which leads to $b = 2ce$. And $c$ is set to 2.

### E. C-perplexity based Weight Reassignment Methods

Similar to X-perplexity based weight reassignment method, we hope the weight of example is still decreasing with its C-perplexity. However, as Fig.2 shows the distribution of C-perplexity, it is highly skewed. If the derivative of the weight of examples is also monotonically decreasing, the weight of examples with C-perplexity greater than 10 will have extreme small weight, which is similar to prune these examples.

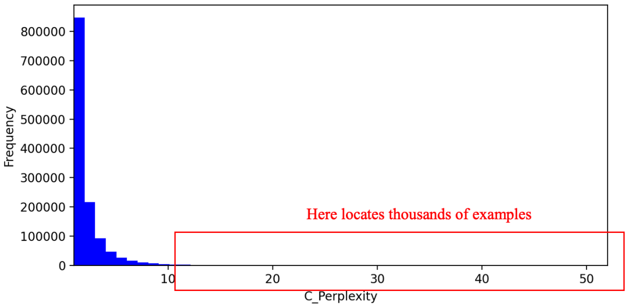

Fig. 2. Distribution C-Perplexity of training image set

Based on the observed distribution, the derivative of the weight function should also decrease with C-perplexity when C-perplexity is small and increase with C-perplexity when it is large. A sigmoid-like function (4) can satisfy these conditions:

$$f(x) = b - \frac{c}{(1 + e^{x-\delta})} \tag{4}$$

Where $b$ and $c$ can control the maximum value and minimum value of weight, and the derivative of weight decreases when $x < \delta$, otherwise increases.

## IV. Experiment

### A. Dataset

Both the training set and validation are from ILSVRC 2012. There are 1.28M training images in the training set with 1,000 categories and 50,000 validation images. In addition, researchers [3] have shown that most of classifiers overfit to existing validation set, and they have developed a novel validation set called Imagenetv2 which all 10,000 examples are sampled from ImageNet. Most of models drop accuracy on this version of validation set. In this project, it is used as another validation set to investigate whether methods improve the generalization ability of the model. To distinguish these validation set, uses validation set I indicates original validation set and validation set II indicates Imagenetv2. For all pruning methods, since the removal of examples will lead the change of class distribution, a random sampling method is adopted to guarantee the class distribution is same to original training set.

### B. Training

All trainings share the same parameters. All models are trained on their dataset using SGD optimizer with momentum 0.8, batch size 256 in 50 epochs. The initial learning rate of the optimizer is 0.01 and it will decay to its 1/10 every 10 epochs.

To control the randomness, all models were initialized as same parameters. And a seed is set.

## V. Result and Analysis

### A. Loss of validation set I

Fig.3 demonstrates the loss of validation set I during epochs. As the legend shown on the upper right corner, different color represents model trained using different methods. All models are converged. The blue line is the loss curve of baseline model. The curve of mislabeled pruning and weight reassignment models are also overlap with the curve of baseline model after 10 epochs. Three threshold pruning models even have larger loss value at the converge stage.

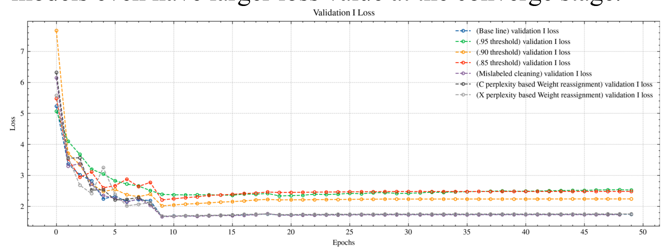

Fig. 3. Loss of validation set I during epochs

### B. Accuracy Table

Table III demonstrates the overall result of each model in training set, validation set I and validation set II. Both Top-1 accuracy and Top-5 accuracy are used as metrics. Furthermore, the different between accuracy of validation set I and validation set II, denoted as "gap", is also reported. The gap indicates the generalization error between validation set I and validation set II. Less gap means the model is less overfitted to the validation set I and may has better generalization ability on other data.

It is unexcepted that all percentile pruning models show worse performance in all metrics. The top-1 and top-5 accuracy on the training set improves a lot. However, since the pruning method removes a portion of examples that a classifier is likely to make wrong prediction, the accuracy of training set seems pointless. The top-1 and top-5 accuracy on validation set I drops with the increase of fraction of pruned examples. For model with threshold 95%, there are 0.06% drop on top-1 accuracy and 0.65% drop on top-5 accuracy. Meanwhile, for model with threshold 85%, there are 0.42% drop on top-1 accuracy and 2.21% drop on top-5 accuracy. It happens on the performance on validation set II. For model with threshold 95%, there are 0.36% drop on top-1 accuracy and 0.13% drop on top-5 accuracy. However, for model with threshold 85%, there are 0.11% drop on top-1 accuracy and 2.34% drop on top-5 accuracy. On the other hand, the performance of 95% threshold pruning model and 90% threshold pruning model show that the gap of top-1 accuracy increases but top-5 accuracy decreases. The situation is reversed for 85% threshold pruning model.

For model using mislabeled pruning method, the top-1 accuracy increases 0.26% and top-5 accuracy increases 0.03% on validation set I. On validation set II, the top-1 accuracy drops 0.11% and top-5 accuracy increase 0.22%. The top-1 gap increases 0.37% the top-5 gap drops 0.19%. The values are all small, compared with those in threshold pruning model.

For models using X-perplexity based and C-perplexity based weight reassignment, both top-1 and top-5 accuracy for validation set I and validation set II increase. The model based on C-perplexity weight reassignment has 0.81% increase of top-1 accuracy on validation set I and 0.52 increase of top-5 accuracy on validation set II. However, it still has 1.03% increase of top-1 gap, which is the largest among all models.

To further investigate the reason that all threshold pruning models have unexcepted drop on accuracy on validation set, a validation set III using all the examples with X-perplexity larger or equal than 0.8 from validation set I is used to conduct the accuracy analysis of all models. The validation III consists of 8,946 examples, about 17.9% of validation set I.

TABLE IV demonstrates the performance of models on the validation set III. Overall, the threshold pruning models show worse performance in classifying difficult examples. The less the threshold, the worse the model. When threshold is set to 95%, there are only 0.06% drop on the top-1 accuracy and 2.47% drop on top-5 accuracy, compared to baseline model. However, the former becomes 1.13% and the latter becomes 7.90% when threshold is 85%. Meanwhile, the mislabeled pruning model and weight reassignment-based models do not show huge different on performance. Some have increase on top-1 accuracy but decrease on top-5 accuracy. Some are the opposite.

make correct prediction. When the threshold is set to 90%, the attention zone moves to the part below the cathedral glass. The model pays more attention on pillars, lights, chairs, desks and people. And it classifies the example as restaurant. When the threshold equals to 85%, it is similar to the 90% threshold pruning model, but the attention zone spreads to irrelevant part of example. However, if the model is based on C-perplexity weight reassignment methods, the attention zone is close to baseline. The Grad-CAM visualization may explain that why threshold pruning models has worse performance on predicting hard examples, as Table IV shows, to some extent.

TABLE III. PERFORMANCE OF MODELS ON TRAING SET, VALIDATION SET I & II

| | Training Set | | Validation I | | Validation II | | Gap (I – II) | |
|---|---|---|---|---|---|---|---|---|
| | Top-1 | Top-5 | Top-1 | Top-5 | Top-1 | Top-5 | Top-1 | Top-5 |
| Baseline | 78.73% | 93.97% | 60.60% | 83.26% | 48.26% | 71.56% | 12.34% | 11.7% |
| .95 Pruning | 87.52% (+8.79%) | 98.23% (+4.26%) | 60.54% (-0.06%) | 82.61% (-0.65%) | 47.90% (-0.36%) | 71.43% (-0.13%) | 12.64% (+0.30%) | **11.18% (-0.52%)** |
| .90 Pruning | 92.90% (+14.17%) | 99.43% (+5.46%) | 60.60% (0%) | 82.02% (-1.24%) | 47.92% (-0.34%) | 70.57% (-0.99%) | 12.68% (+0.34%) | 11.45% (-0.25%) |
| .85 Pruning | 96.11% (+17.38%) | 99.81% (+5.84%) | 60.18% (-0.42%) | 81.05% (-2.21%) | 48.15% (-0.11%) | 69.22% (-2.34%) | **12.03% (-0.31%)** | 11.83% (+0.13%) |
| Pruning (Mislabeled) | 78.98% (+0.25%) | 94.10% (+0.13%) | 60.86% (+0.26%) | 83.29% (+0.03%) | 48.15% (-0.11%) | 71.78% (+0.22%) | 12.71% (+0.37%) | 11.51% (-0.19%) |
| Weight (X) | 80.31% (+1.58%) | 93.92% (-0.05%) | 61.23% (+0.63%) | **83.63% (+0.37%)** | **48.68% (+0.42%)** | 71.96% (+0.40%) | 12.55% (+0.21%) | 11.67% (-0.03%) |
| Weight (C) | 80.19% (+1.46%) | 93.71% (-0.26%) | **61.41% (+0.81%)** | 83.42% (+0.16%) | 48.48% (+0.22%) | **72.08% (+0.52%)** | 13.37% (+1.03%) | 11.34% (-0.36%) |

TABLE IV. PERFORMANCE OF MODELS ON VALIDATION SET III

| Model | Validation set III | |
|---|---|---|
| | Top-1 | Top-5 |
| Baseline | 4.88% | 38.01% |
| .95 Pruning | 4.82% (-0.06%) | 35.54% (-2.47%) |
| .90 Pruning | 4.33% (-0.55%) | 33.22% (-4.79%) |
| .85 Pruning | 3.57% (-1.31%) | 30.11% (-7.90%) |
| Pruning (Mislabeled) | 4.85% (-0.03%) | 38.05% (+0.04%) |
| Weight (X) | 5.21% (+0.33%) | 37.82% (-0.19%) |
| Weight (C) | 4.63% (-0.25%) | 38.20% (+0.19%) |

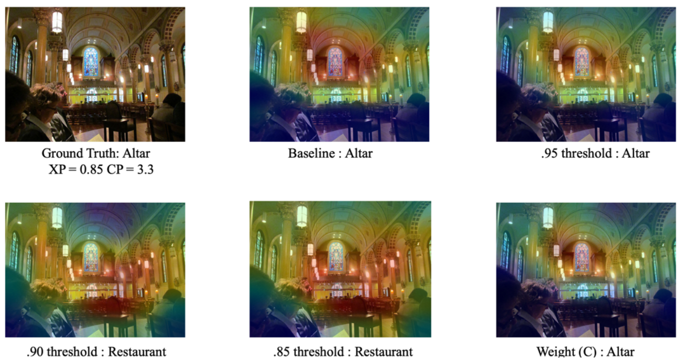

Fig. 4. Grad-CAM of an example belongs to Altar

### C. *Grad-CAM visualization*

Grad-CAM [5], or Gradient-weighted Class Activation Mapping, it is an effective visualization to show which part of image the model pays more attention on when making decisions. It uses the feature map in the last convolutional layer to draw the heat-map to show the importance of pixels.

Fig.4 demonstrates the Grad-CAM of baseline, three percentile pruning models, and model based on C-perplexity weight reassignment methods while predicting the examples with ground truth altar. The X-perplexity of the example is 0.85 and the C-perplexity and 3.3. When predicting this example, baseline model pays much more attention of the cathedral glass, along with the dome. For the Gram-CAM of 95% threshold pruning model, the weight of this area becomes smaller, or less obvious. But the model can still

### D. *Accuracy of classes with different difficulty*

The analysis based on classes with different difficulty is also conducted. The difficulty is defined as X-perplexity. 50 classes are selected with average X-perplexity 0.11, called easy-level classes; 50 classes are selected with average X-perplexity 0.42, called mid-level classes; and 50 classes are selected with average X-perplexity 0.71, called hard-level classes. One particular class can only belong to one class.

Fig.5 demonstrates the average accuracy on each level class. Color orange represents easy-level classes, blue represents mid-level classes, and green represents hard-level classes. three auxiliary lines are drawing to help to compare. There is no difference among the performance of each model on classifying easy-level images. Models except 95% threshold pruning show greater accuracy on classifying easy-level images. The case is similar for mid-level classes. All models except 85% threshold pruning have exactly the same accuracy on classifying mid-level class. For hard-level classes, 90% threshold pruning and 85% threshold pruning model have lower accuracy, which is consistent with the situation in Table III, Table IV, and Fig. 4. Other models have higher accuracy than baseline model.

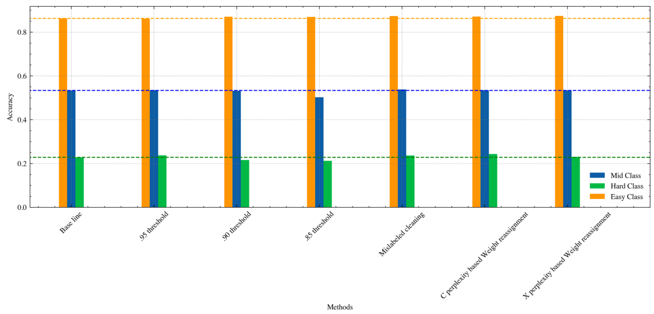


Fig. 5. Avg. accuracy on each level class

## *E. Example-based analysis*

To Further investigate the impact of different methods on training, three examples with varies X-perplexity are selected: an example labeled as cup with X-perplexity 0.03, an example labeled as pirate with X-perplexity 0.41, and an example labeled as rock crab with X-perplexity 1.0.

As Fig.6 shows, for each example, a probability bar plot and an entropy bar plot for each model are drawn. The probability bar plot shows the probability that the given example is classified to the ground truth label and the entropy bar plot shows the entropy of the logit vector. In other word, the less the entropy, the higher the confidence of the model on this example.

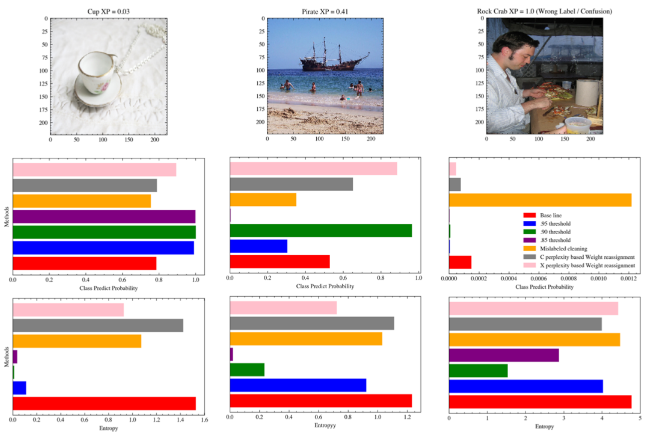


Fig. 6. Probability bar plot and Entropy bar plot of examples with varies X-perplexity

For the cup example, models based on weight reassignment or mislabeled cleaning have close accuracy to baseline model. Three threshold pruning models have almost 20% larger accuracy. And the entropy of each model is lower than baseline model. Three threshold pruning models have extremely small entropy, which means they are highly confident on classifying this example.

The case is slightly different for example pirate. Two weight reassignment models and 90% threshold pruning model demonstrate higher accuracy on classifying this example while mislabeled pruning model and 95% threshold pruning model drop a portion of accuracy. However, 85% threshold pruning model fails to predict this example and the probability is extremely low. Compared to base line model, all models have lower entropy of the logit vector. Particularly, the 85% threshold pruning model has extremely low entropy. In other word, it has strong confidence on classifying this example as other classes, instead of pirate.

The rock crab image shows a man is cutting a crab. However, the crab is crushed, and we have no idea whether it is a rock crab. It is difficult for even human to classify it correctly. That explains the reason that the X-perplexity is 1.0, i.e., no model makes correct prediction. The prediction probability that baseline model gives is 0.015%, except mislabeled pruning model has eight times larger probability (around 0.12%), all models have lower probability. And all models have lower entropy.

# VI. CONCLUSION

Example perplexity is used to describe the extent of difficulty of classifying. X-perplexity is defined as the fraction of models that make wrong predictions over the population model and C-perplexity is defend as the geometric mean of perplexity of the probability distribution over $N$ models. An intuitive idea to optimize the training set according to perplexity to improve the performance of models, which is also the purpose of this project.

In the project, several methods are adopted. First, pruning the examples with perplexity larger than a certain threshold. Three thresholds: 95% percentile, 90% percentile, and 85% percentile are used to prune example. Furthermore, inspired by the characteristic of mislabeled examples, a mislabeled pruning method is adopted to prune example with high X- perplexity and low C-perplexity. In addition, weight reassignment methods based on X-perplexity and C-perplexity are also adopted.

Several interested findings are found. First, the performance of models is not improved as expected. on the contrary, the accuracy on validation set drops using percentile pruning methods. According to further experiments, percentile pruning methods lead to worse generalization ability on hard examples while other methods remain the same. The Grad-CAM visualization gives a possible explanation that the percentile pruning method transfer the attention zone of the example. Those models fail to focus on area where they should focus on, and it led to wrong prediction. This does not happen on other models. To find the potential impact on accuracy of classes with different X-perplexity, three groups with 50 class per group with X-perplexity 0.11, 0.42, and 0.71 are used to examine the accuracy of models on predicting examples within groups. The result suggests that basically all models have same performance on classifying easy-level and mid-level classes, but 90% percentile pruning and 85% percentile pruning models' loss accuracy on predicting hard-level example. The probability bar plots, and entropy bar plots show that all models tend to give lower entropy when predicting an example and higher accuracy on predicting easy examples and mid-level example. For examples with high X-perplexity, the probability of models is lower than baseline model.

The result of project suggests that threshold pruning methods have negative impact. They loss accuracy on validation set especially for those examples have high X-perplexity. It probably due to a large faction of "difficult examples" are pruned in the training set, which destroys the structure of data. Mislabeled pruning and weight reassignment may have Positive impact on the model.

# REFERENCES


[1] Nevin L. Z , Weiyan X., Zhi L, et al. Example Perplexity.

[2] Deng J, Dong W, Socher R, et al. Imagenet: A large-scale hierarchical image database[C]//2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009: 248-255.

[3] Recht, B., Roelofs, R., Schmidt, L., & Shankar, V. (2019, May). Do imagenet classifiers generalize to imagenet?. In International Conference on Machine Learning (pp. 5389-5400). PMLR.

[4] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4510-4520).

[5] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision (pp. 618-626).

## APPENDIX A: MINUTES

Minutes of 1st Meeting

Date: 2021-09-08
Time: 6:00 pm
Place: Zoom
Attending: QIU Yaowen, SHEN Yiqin, Anthony Chiu, and all students applied for Prof. Nevin L.Zhang's project
Supervisor: Prof. Nevin L.Zhang

1. Approval of minutes
   This is first formal group meeting, so there were no minutes to approve
2. Discussion Items
   - Administrative matters are announced
   - Time schedule is announced
   - Basic introduction of project
   - Q & A
3. Goals for next meeting
   Each student should give a clear description of project objective and scope

Minutes of 2nd Meeting

Date: 2021-09-29
Time: 3:00 pm
Place: Zoom
Attending: QIU Yaowen, SHEN Yiqin, Anthony Chiu
Supervisor: Prof. Nevin L.Zhang

4. Approval of minutes
   The minutes of the last meeting were approved without amendment
5. Discussion Items
   - Each student gave a clear description of project objective and scope
   - The supervisor gave advice to each student and answer student's questions
6. Goals for next meeting
   Each student should give a progress report of meeting

Minutes of 3rd Meeting

Date: 2021-10-27
Time: 3:00 pm
Place: Zoom
Attending: QIU Yaowen, SHEN Yiqin, Anthony Chiu
Supervisor: Prof. Nevin L.Zhang

7. Approval of minutes
   The minutes of the last meeting were approved without amendment
8. Discussion Items
   - Each student reported the progress of project
   - The supervisor asked questions about the detail of the progress and gave advice
9. Goals for next meeting
   Each student should give a final project report

Minutes of 4th Meeting

Date: 2021-12-03
Time: 3:00 pm
Place: Zoom
Attending: QIU Yaowen, SHEN Yiqin, Anthony Chiu
Supervisor: Prof. Nevin L.Zhang

10. Approval of minutes

The minutes of the last meeting were approved without amendment

11. Discussion Items
    - Each student gave final report of the whole project
    - The supervisor asked questions in detail
12. Goals for next meeting
    N/A