NYCU Introduction to Machine Learning, Homework 4

111550035, 蔡昀錚

Part. 1, Kaggle (70% [50% comes from the competition]):

(10%) Implementation Details

Here are the approaches that I use for the best model:

1. Manual data preprocessing: I remove the image in training data which is not relevant to the class. Example: (Surprise/353.jpg)

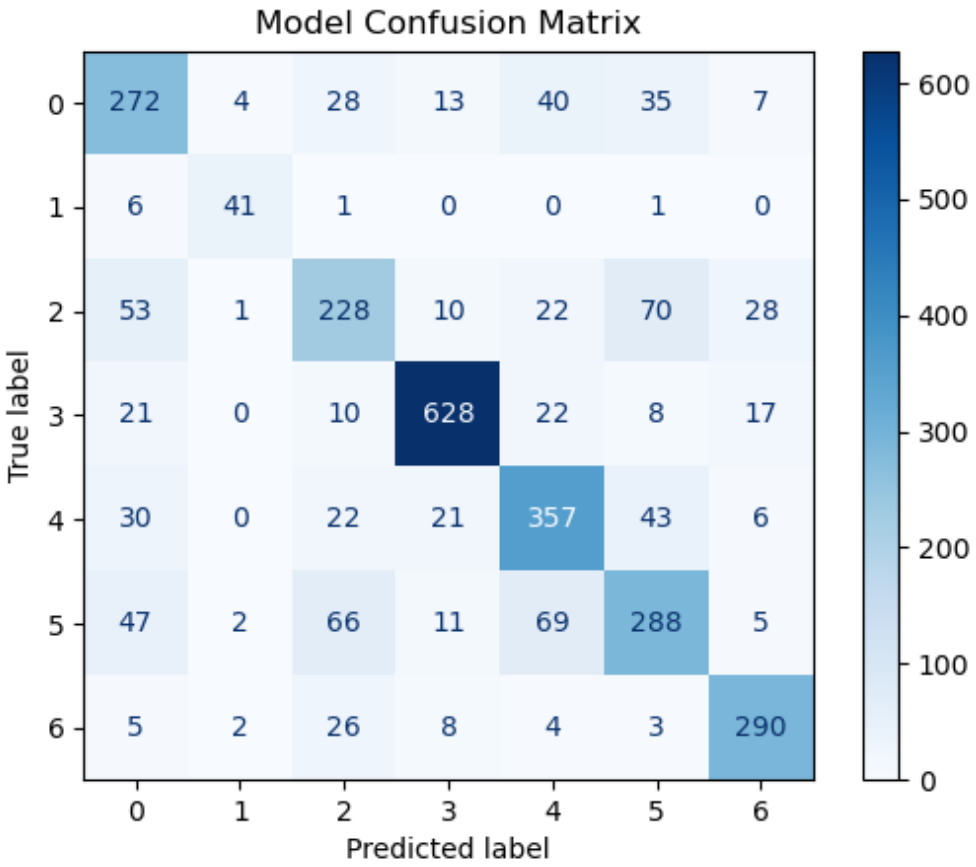
2. I apply several data augmentation methods:
   (1) Flipped image
   (2) Gaussian noise
   (3) Rotation
   (4) Random Crop
3. I apply ensemble method to several model to improve the performance. I ensemble:
   (1) ResNet18 pretrained weights + fine-tuning
   (2) ResNet18 pretrained weights + flip / Gaussian noise data augmentation
   (3) ResNet18 pretrained weights + rotation data augmentation
   (4) ResNet50 pretrained weights + data augmentation
   Each has about 56~58% accuracy (based on Kaggle), ensemble them to get 61% accuracy. In Addition, although these models have similar scale output, I normalized the output of each model to make them have same influence on the final result.

Model backbone (e.g., VGG16, VGG19, Custom, etc)	ResNet18 (Pt: ResNet18_Weights.IMAGENET1K_V1), ResNet50 (Pt: ResNet50_Weights.IMAGENET1K_V2)
Number of model parameters	11689512+25557032=37246544=37M acc@1 (on ImageNet-1K) 69.758 acc@5 (on ImageNet-1K) 89.078 min_size height=1, width=1 categories tench, goldfish, great white shark, ... (997 omitted) num_params 11689512 recipe link GFLOPS 1.81 File size 44.7 MB acc@1 (on ImageNet-1K) 80.858 acc@5 (on ImageNet-1K) 95.434 min_size height=1, width=1 categories tench, goldfish, great white shark, ... (997 omitted) num_params 25557032 recipe link GFLOPS 4.09 File size 97.8 MB
Other hyperparameters …	(AdamW optimizer) Learning rate = 0.001 Weight decay = 1e-4 Gamma = 0.9

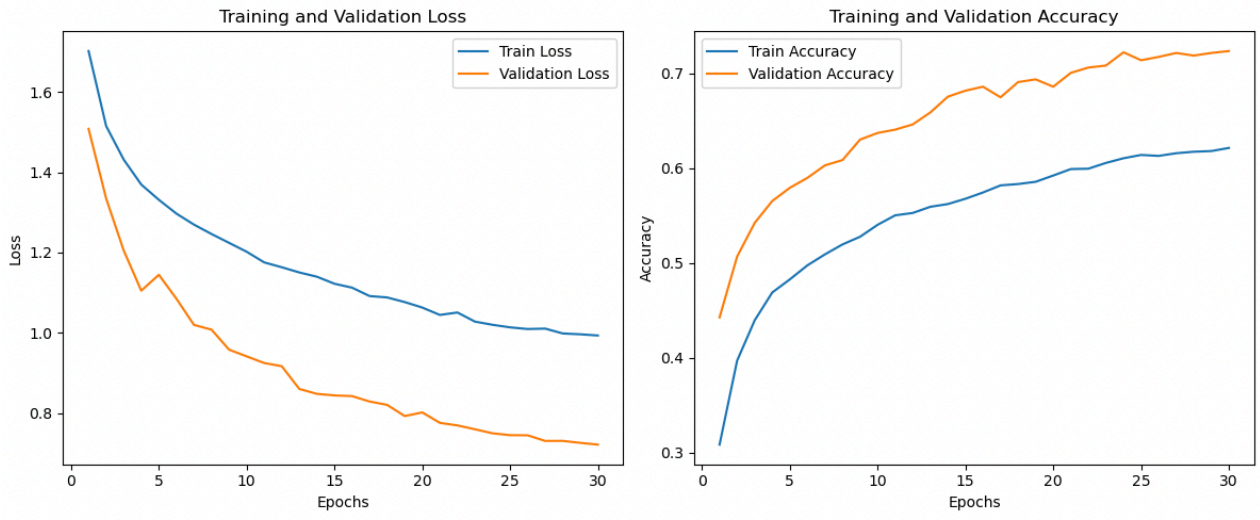
(10%) Experimental Results

1. Evaluation metrics and learning curve

I separate a validation set (10% of the training data) as the evaluation metric.



Confusion matrix



Learning curve

2. Ablation Study

Here are approaches that I tried to improve my model:

Method	Performance (Acc)	My opinion
Data augmentation - rotation	Increase ~1%	Rotation may prevent the model from overfitting, especially small scale dataset. But rotation might also produce black border that will affect the model, so the performance with and without rotation is similar.
Data preprocessing	-	I do histogram equalization to training and testing dataset, since I find a portion of the dataset is either too bright or too dark. However the accuracy doesn't improve, this may due to the fact that histogram equalization might loss the information and feature of images.
Freeze	Decrease ~2%	Since ResNet is a big model, while I only have small training set, I freeze some layer of ResNet18 (with pretrained weights ImageNet).
Ensembling (Voting)	Increase ~5%	Ensemble the models is an efficient way to improve the performance, but I need to have high variation between each model by bootstrapping or different architecture. However I only have small scale dataset and few kinds of model architecture, so the improvement is not that significant.
Balance dataset	Decrease ~2%	I find that the data size of each class has a large difference. Thus, I increase the influence to model of the samples of smaller classes. However I didn't get a better result, this may due to the fact that only small portion of data dominant the model weights, which may result in bad performance such like overfitting.
Bagging	Decrease a lot	Bagging should be a good approach of this model, however with limited time and colab's limitation, I can just run with 5 classifier each with 10 epoch, so the result is bad due to underfitting.

Part. 2, Questions (30%):

1. (10%) Explain the support vector in SVM and the slack variable in Soft-margin SVM. Please provide a precise and concise answer. (each in two sentences)

   (1) Support Vector: support vectors are the data points closest to the decision boundary and directly influence w. These points are critical because the margin is maximized based on their distances to the hyperplane.

   (2) Slack Variable: In soft-margin SVM, slack variables measure the extent to which a data point violates the margin constraints. They allow the model to handle non-linearly separable data by permitting some misclassification or margin violations, controlled by regularization parameter C.

2. (10%) In training an SVM, how do the parameter C and the hyperparameters of the kernel function (e.g., γ for the RBF kernel) affect the model's performance? Please explain their roles and describe how to choose these parameters to achieve good performance.

   (1) C controls the balance between maximizing the margin and minimizing classification errors.
      1. Large C: prioritizes minimizing classification errors, which can lead to a smaller margin and potentially overfitting the training data.

2. Small C: allows more margin violations, promoting a larger margin and better generalization but may underfitting.

(2) Kernel function (γ for RBF kernel) determines the influence of a single training sample on the decision boundary.
1. Large γ: focuses on nearby points, creating highly flexible decision boundaries but increasing the risk of overfitting.
2. Small γ: considers distant points, resulting in smoother, less flexible boundaries that may underfit the data.

(3) Choosing Policy:
1. Perform a grid search over a range of C and γ. Then use k-fold cross-validation to evaluate the performance of each combination, choosing the pair that balances accuracy and generalization.
1. Plot accuracy against C and γ.Then identify regions where the model avoids overfitting or underfitting.

3. (10%) SVM is often more accurate than Logistic Regression. Please compare SVM and Logistic Regression in handling outliers.

(1) SVM is robust to outliers due to its focus on maximizing the margin based on support vectors, which are typically not outliers. However, if C is set too high in soft-margin SVM, it may try to perfectly classify all points, leading to overfitting, including being influenced by outliers.

(2) Logistic Regression is sensitive to outliers because the loss function (log loss) penalizes misclassified points heavily, regardless of their distance from the decision boundary.

(3) Difference:
- SVM with a properly tuned C and kernel can mitigate the effect of outliers by allowing margin violations.
- Logistic Regression inherently lacks a mechanism to ignore or down-weight outliers without additional preprocessing or modifications.