STAT433 Final Project

ArchEnsemble: Architectural Approach for More Efficient Ensemble

Hyogon Ryu

Department of Cyber Defense, Korea University

2019330023

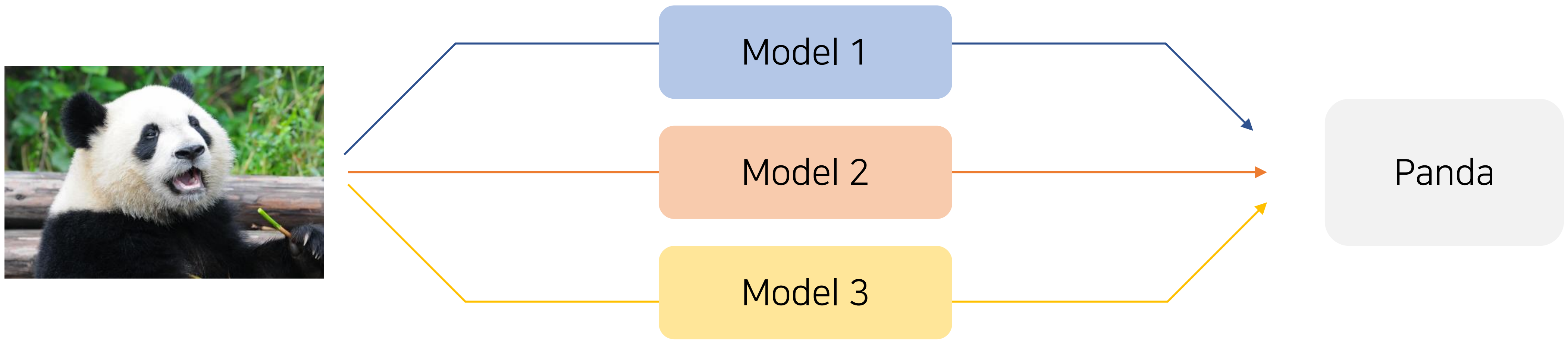
Contents

- Why this research?
- Related Works
- Approach
- Experiments
- Conclusion

Why this research?

Ensemble in Deep Neural Network

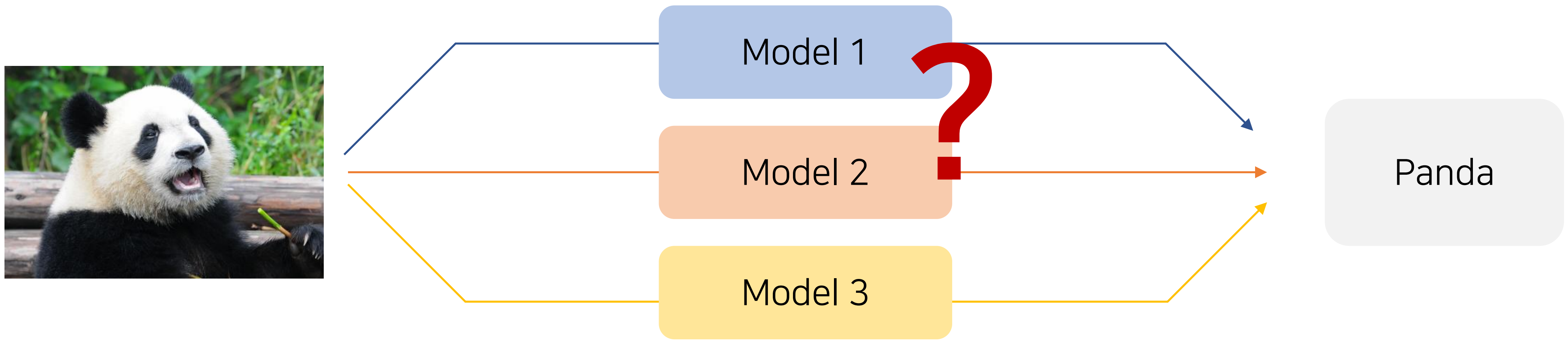
- Model Ensemble's Pros
  - Higher Accuracy & Robustness
- Model Ensemble's Cons
  - Standard Ensemble Method needs the multiple computation cost.
  - Especially Deep Neural Network, computation cost is the important issue.



Why this research?

Ensemble in Deep Neural Network

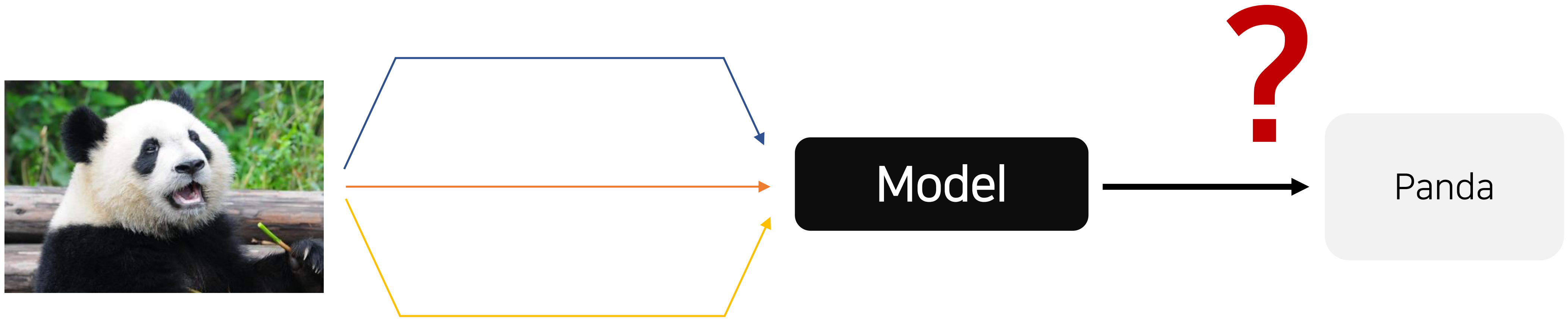
- Model Ensemble's Pros
  - Higher Accuracy & Robustness
- Model Ensemble's Cons
  - Standard Ensemble Method needs the multiple computation cost. (training time / inference time)
  - Especially Deep Neural Network, computation cost is the important issue.



Why this research?

Ensemble in Deep Neural Network

- Model Ensemble's Pros
  - Higher Accuracy & Robustness
- Model Ensemble's Cons
  - Standard Ensemble Method needs the multiple computation cost. (training time / inference time)
  - Especially Deep Neural Network, computation cost is the important issue.



Related Works

papers

- Batchensemble: an alternative approach to efficient ensemble and lifelong learning
- Learning Loss for Test-Time Augmentation

Related Works

Batchensemble: an alternative approach to efficient ensemble and lifelong learning

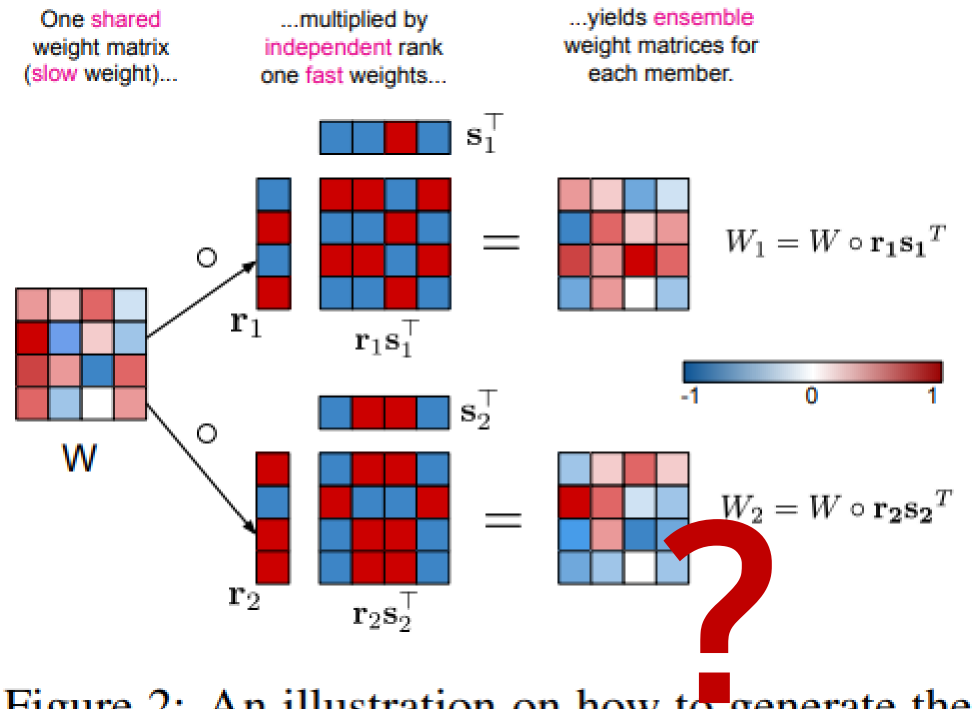
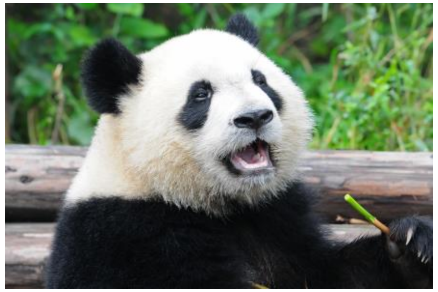
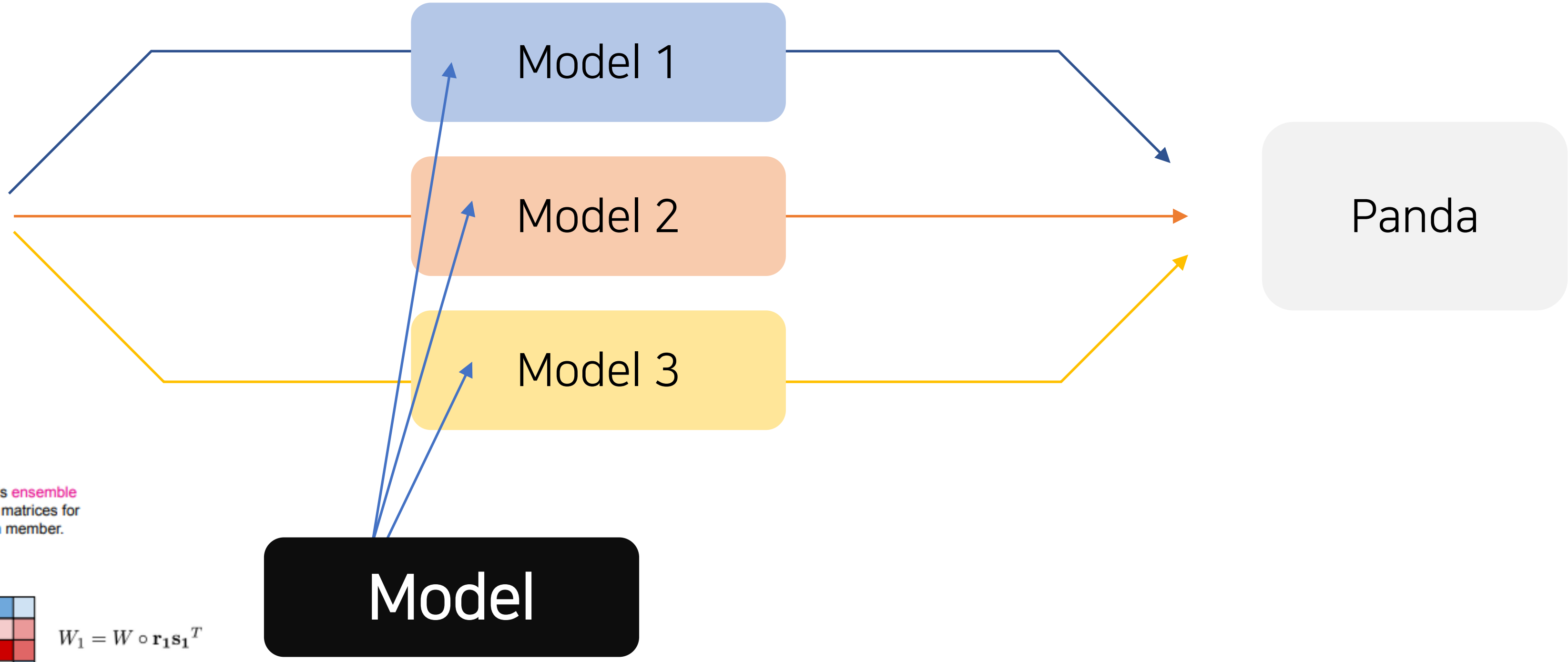


Figure 2: An illustration on how to generate the ensemble weights for two ensemble members.

Related Works

Batchensemble: an alternative approach to efficient ensemble and lifelong learning





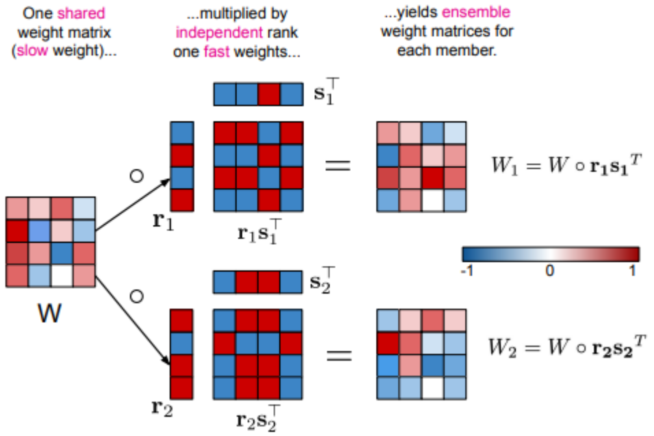


Figure 2: An illustration on how to generate the ensemble weights for two ensemble members.

Related Works

Learning Loss for Test-Time Augmentation

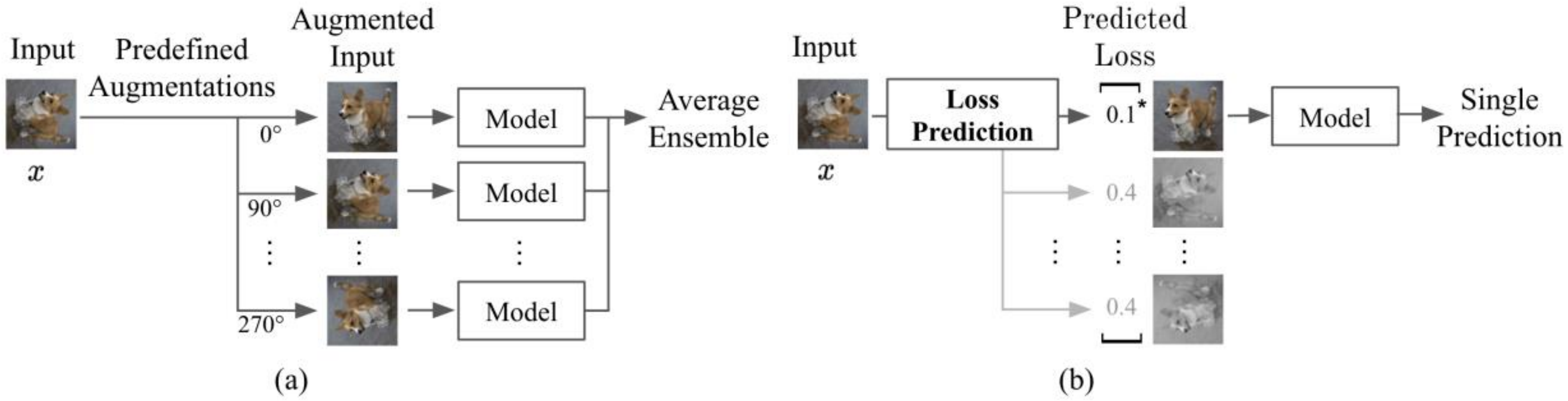
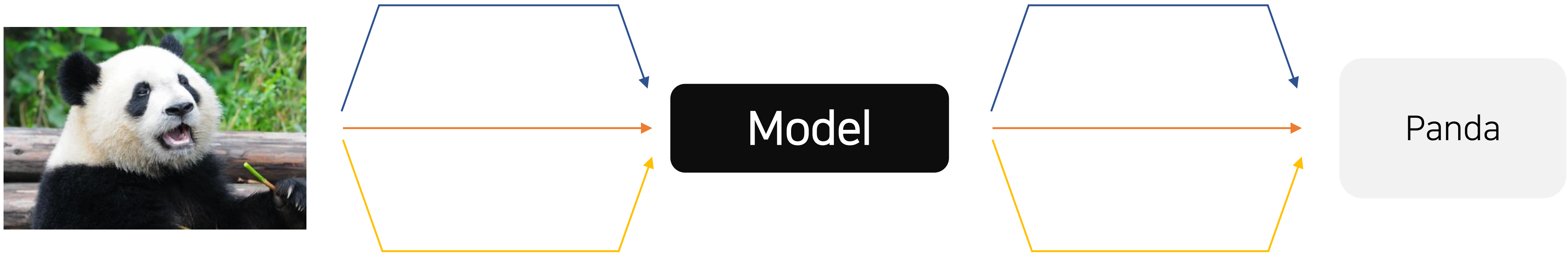


Figure 1: Conceptual comparison between conventional test-time augmentation and the proposed test-time augmentation. (a) Conventional test-time augmentation. (b) Our proposed test-time augmentation. Previous test-time augmentations use prefixed transformations regardless of input. On the other hand, our method predicts the loss value for each transformation before choosing one or a few. Note that this figure shows only one augmentation is selected by predicted losses, i.e. $k = 1$.

Approach

Main idea

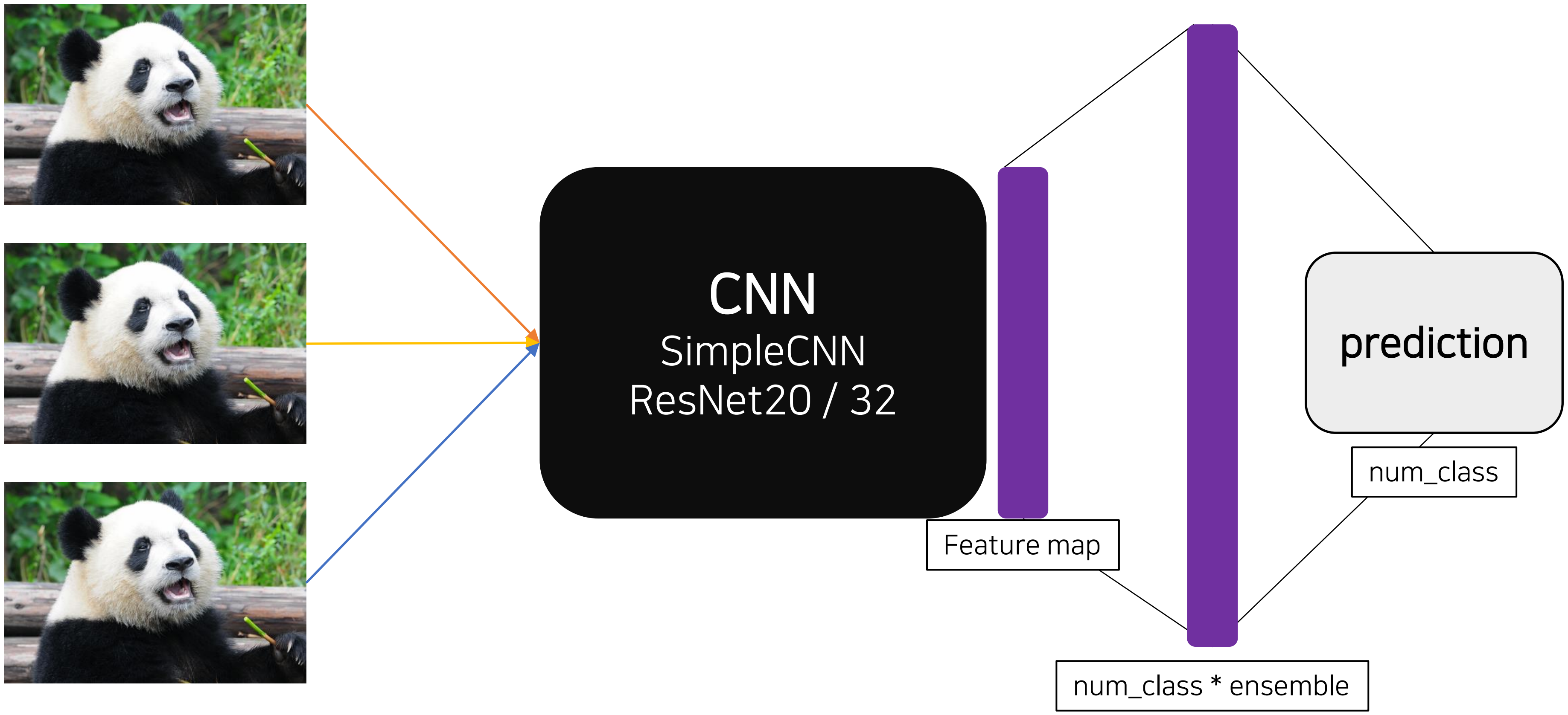
- Is there a model architecture that makes an ensemble-like effect?



Approach

Main idea

- Is there a model architecture that makes an ensemble-like effect?

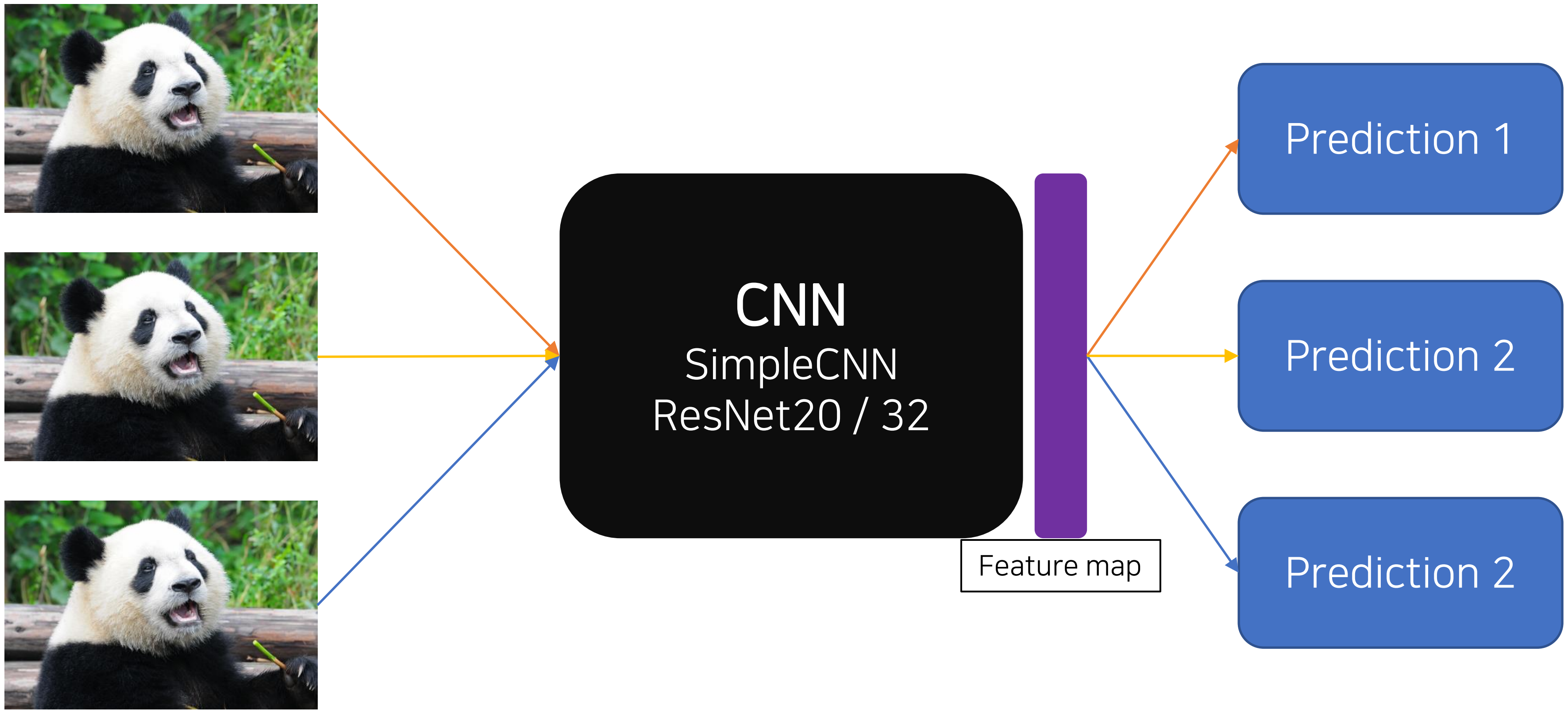


ArchEnsemble: Architecture

Approach

trial and error

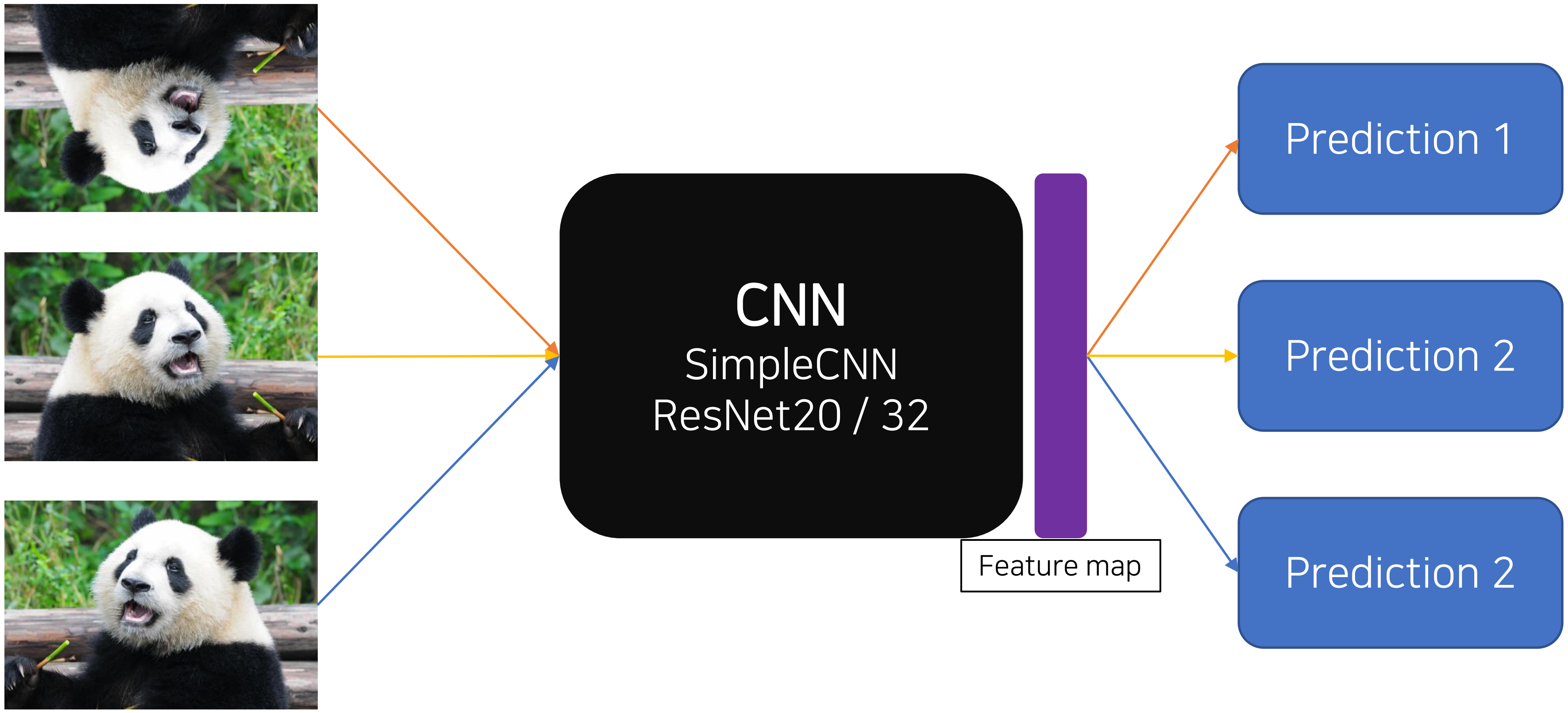
- Is there a model architecture that makes an ensemble-like effect?



Approach

trial and error

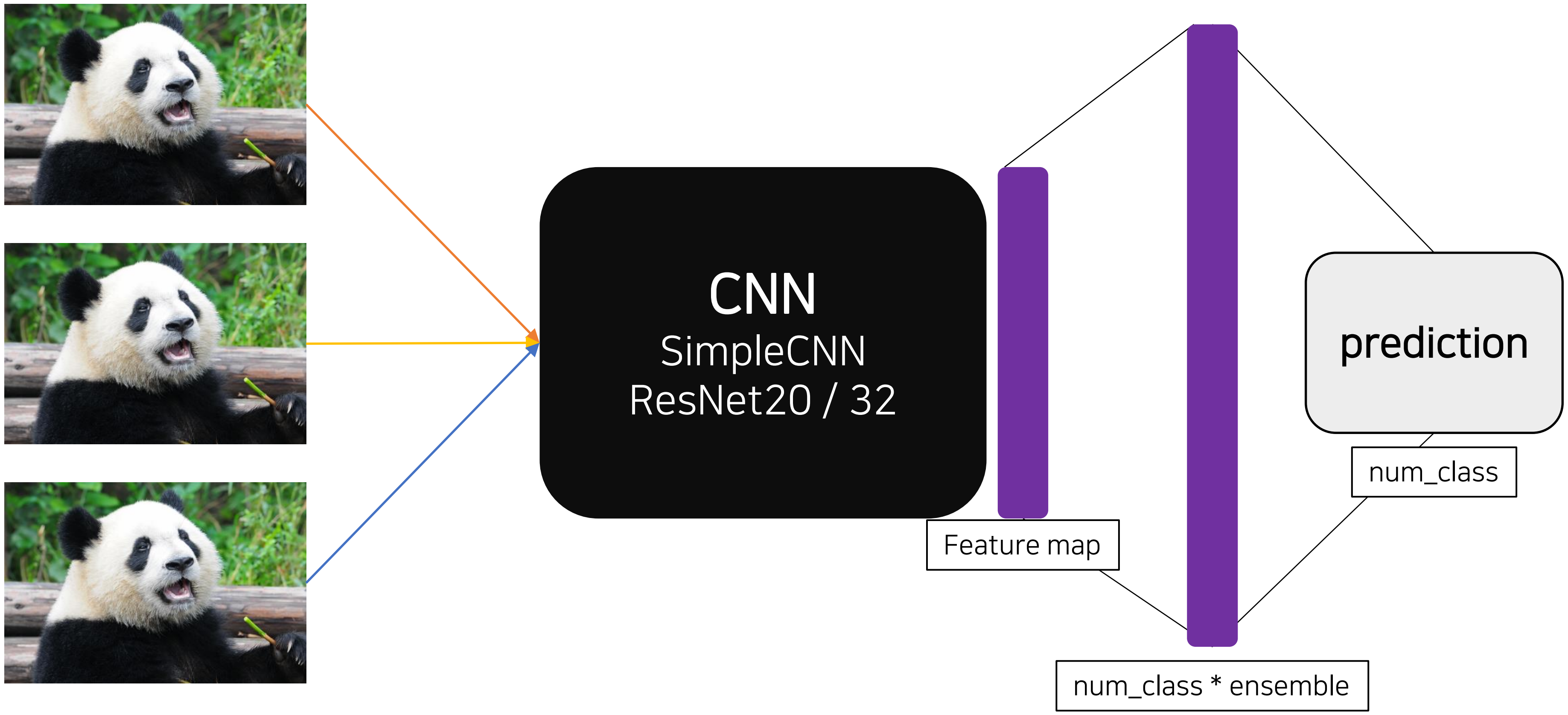
- Is there a model architecture that makes an ensemble-like effect?



Approach

Main idea

- Is there a model architecture that makes an ensemble-like effect?

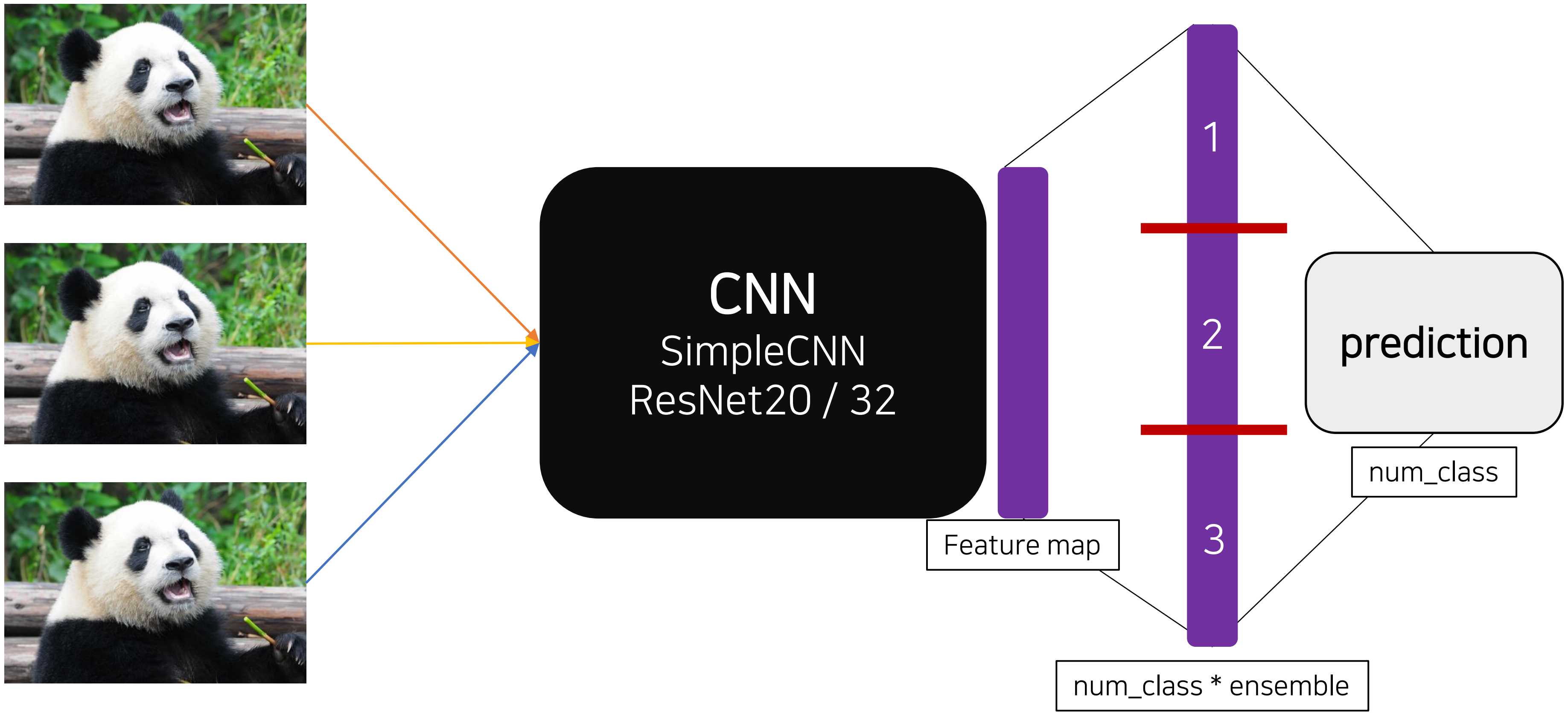


ArchEnsemble: Architecture

Approach

Main idea

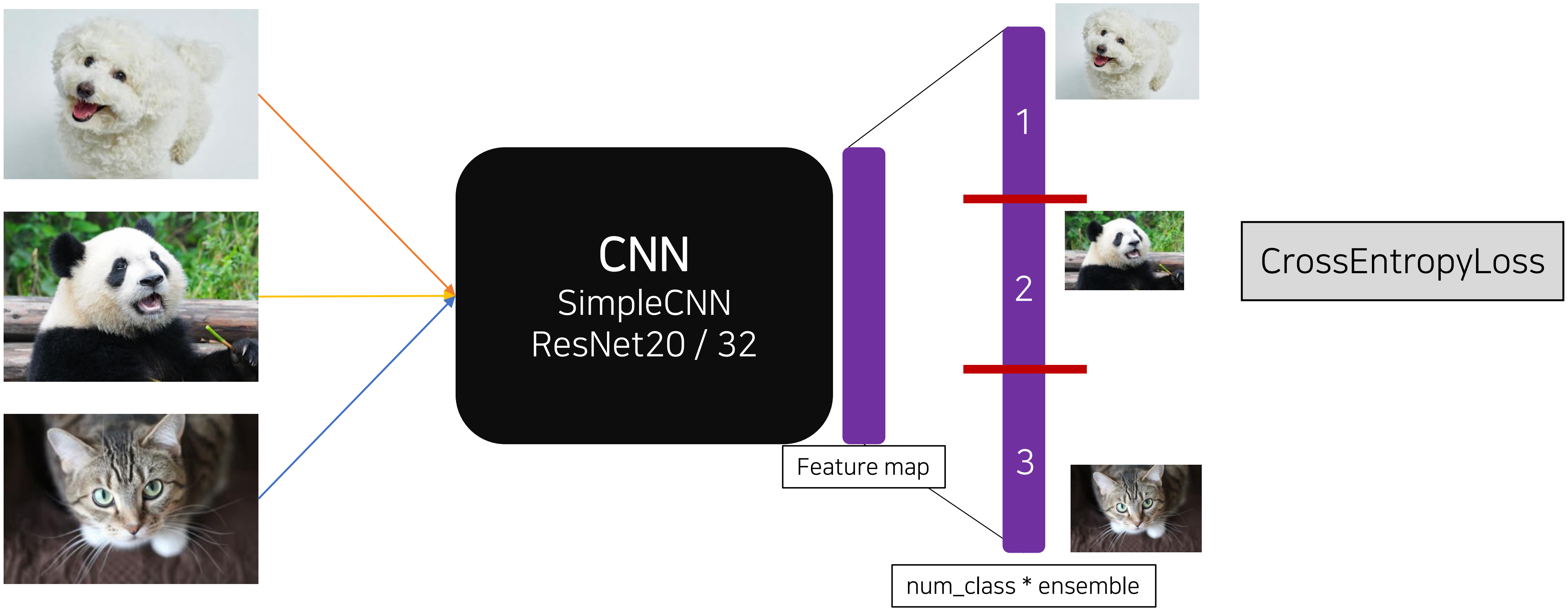
- Is there a model architecture that makes an ensemble-like effect?



ArchEnsemble: Architecture

Approach

How to Train?



ArchEnsemble: How to Train?

Experiments

Setting

Since this is not the application project,

I didn't work hard to optimize the hyperparameter

Exp. on Colab.

- Dataset: CIFAR10, CIFAR100
- Backbone CNN Model:
  SimpleCNN(2 conv-layer), ResNet20, ResNet32
- ensemble : 1, 2, 4

- Optimizer : Adam
- Learning Rate Schedular: StepLR
- Learning rate: 1.0
- Learning rate step gamma : 0.7

- batch_size 10

Experiments

Result

- SimpleCNN

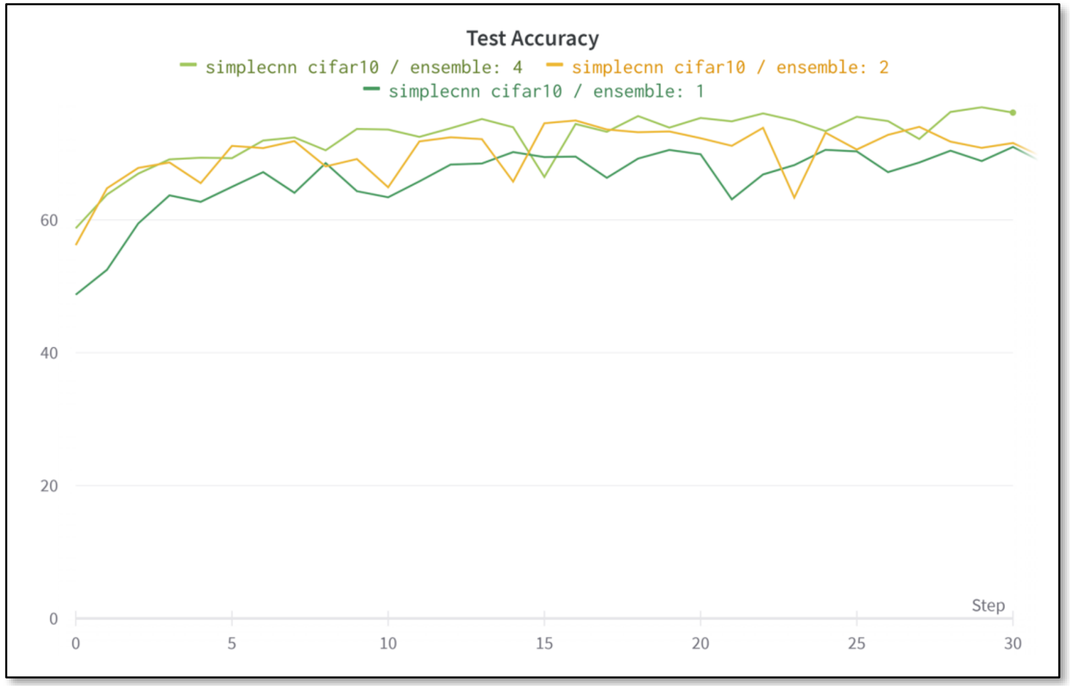


Fig1. CIFAR-10/SimpleCNN test accuracy(ensemble: 1,2,4)

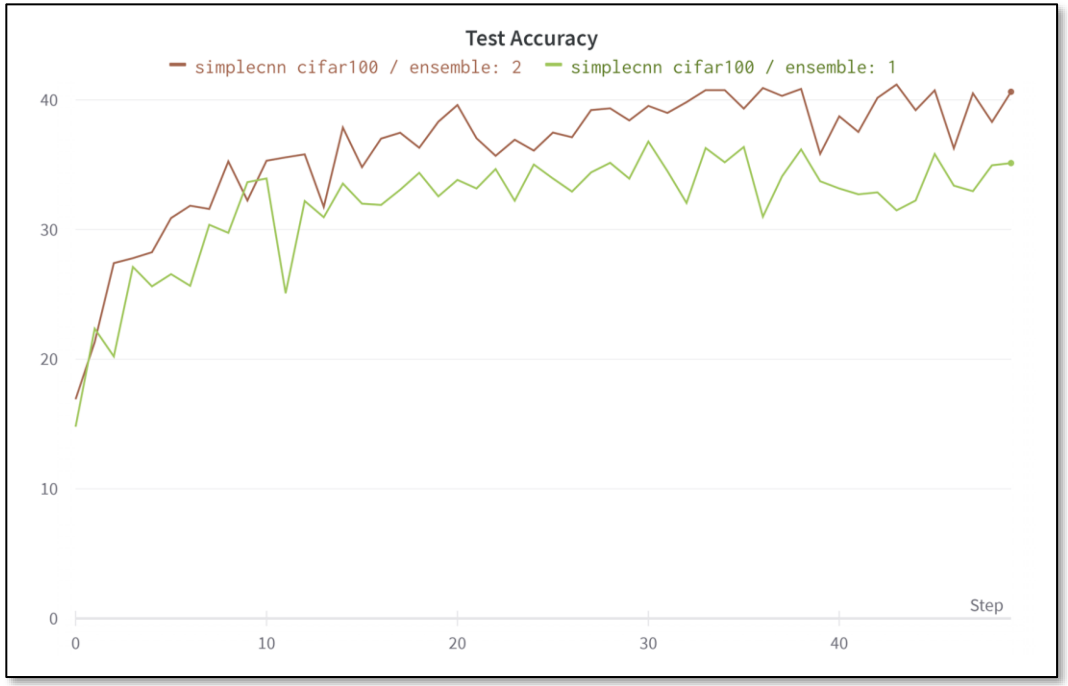


Fig2. CIFAR-100/SimpleCNN test accuracy(ensemble: 1,2)

Experiments

Result

- ResNet20

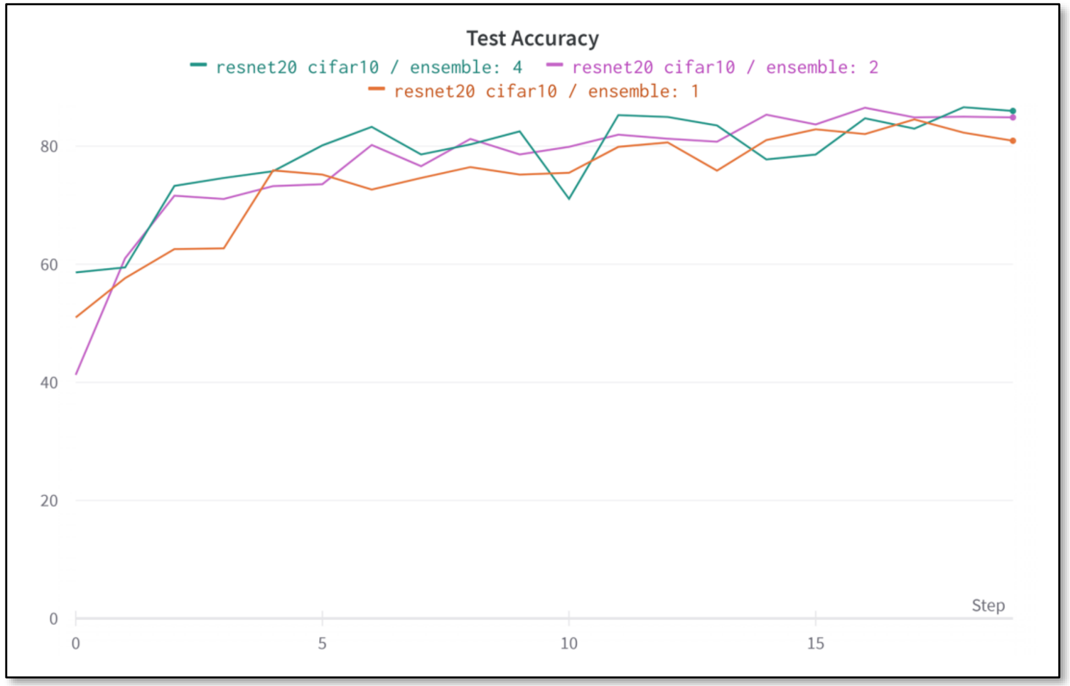


Fig3. CIFAR-10/ResNet20 test accuracy(ensemble: 1,2,4)

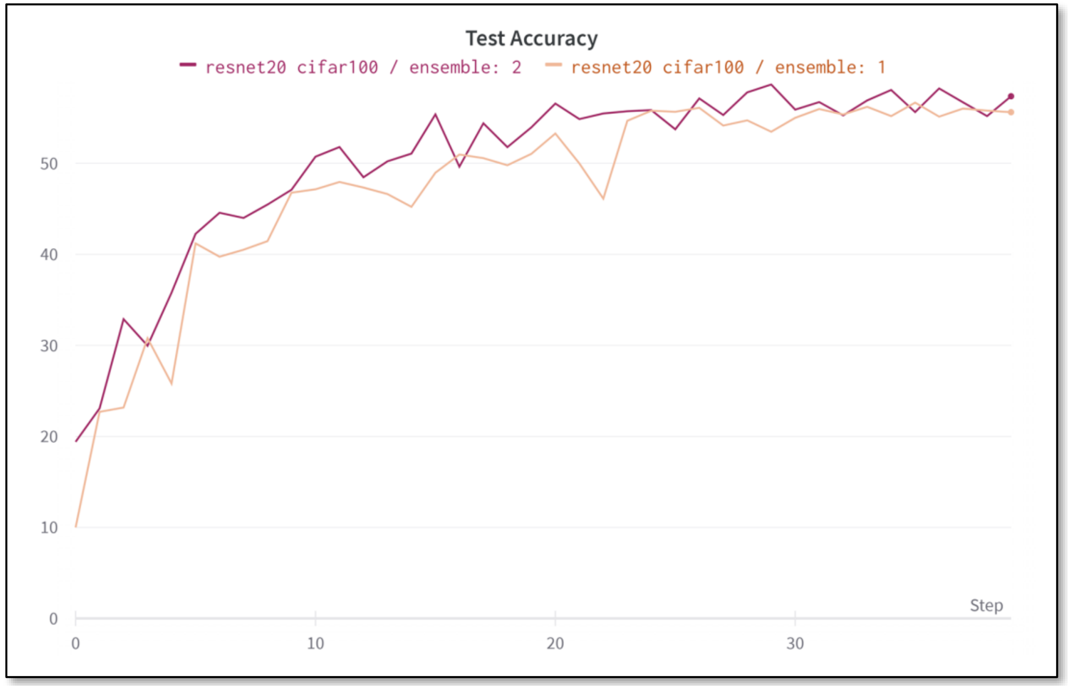


Fig4. CIFAR-100/ResNet20 test accuracy(ensemble: 1,2)

Experiments

Result

- ResNet32

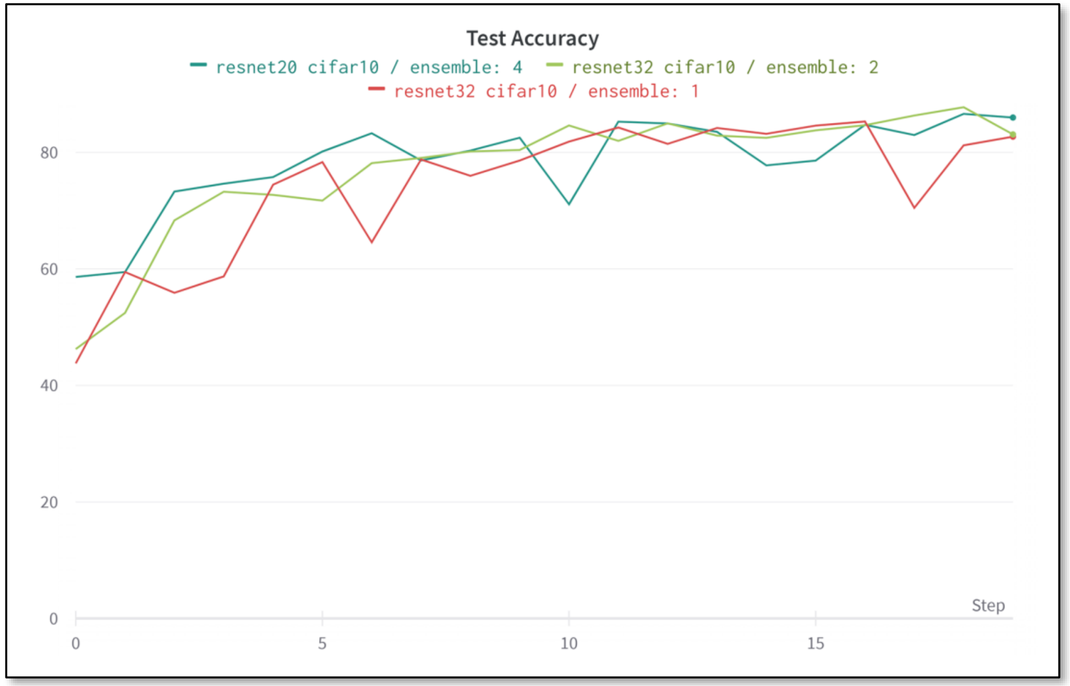


Fig5. CIFAR-10/ResNet32 test accuracy(ensemble: 1,2,4)

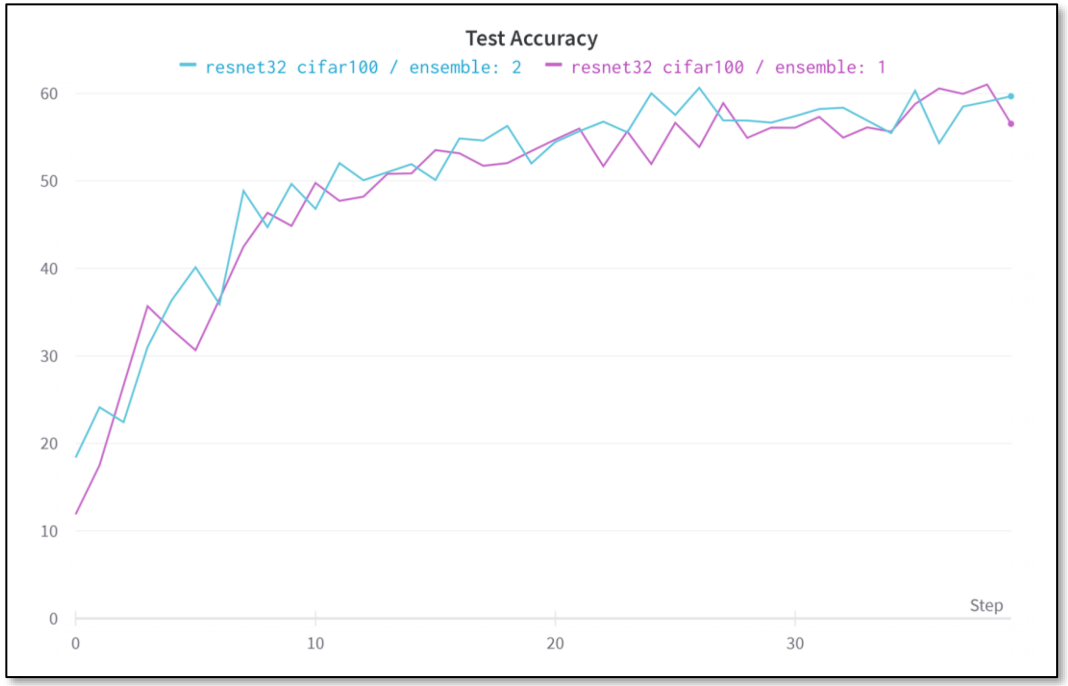


Fig6. CIFAR-100/ResNet32 test accuracy(ensemble: 1,2)

Experiments

Result

Model	# of Ensemble	CIFAR-10	CIFAR-100
SimpleCNN	1	68.63	35.13
	2	74.35	**40.63**
	4	**76.15**	
ResNet20	1	80.92	52.03
	2	84.88	**56.46**
	4	**85.97**	
ResNet32	1	82.69	56.53
	2	83.07	**59.68**
	4	**87.02**	

Tables 1. top-1 Accuracy of each model. Best results are Highlighted

Conclusion

conclusion

- I propose the ArchEnsemble, a novel architectural efficient ensemble method.
- It's very simple and easy to implement.
- In addition, this method can be applied to all kind of CNNs.
- This method doesn't have to train the other model. It's greatly benefited in computation cost.
- By applying this method, model accuracy can be significantly increased.

- It worked well for a simple architecture, but the more complex the architecture, the worse the performance. This part needs analysis.

reference

- Wen, Y., Tran, D., & Ba, J. (2020). Batchensemble: an alternative approach to efficient ensemble and lifelong learning. arXiv preprint arXiv:2002.06715.
- Kim, I., Kim, Y., & Kim, S. (2020). Learning loss for test-time augmentation. Advances in Neural Information Processing Systems, 33, 4163-4174.
- Soflaei, M., Guo, H., Al-Bashabsheh, A., Mao, Y., & Zhang, R. (2020, April). Aggregated learning: A vector-quantization approach to learning neural network classifiers. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 04, pp. 5810-5817).
- Krizhevsky, A., & Hinton, G. (2009). Learning multiple layers of features from tiny images.
- https://github.com/akamaster/pytorch_resnet_cifar10