Bag of Tricks for Learning from Web Data

Lin Chen, Zhikun Lin, Anyin Song, Yaxiong Chi,
Chenhui Qiu, Shouping Shan, Lixin Duan

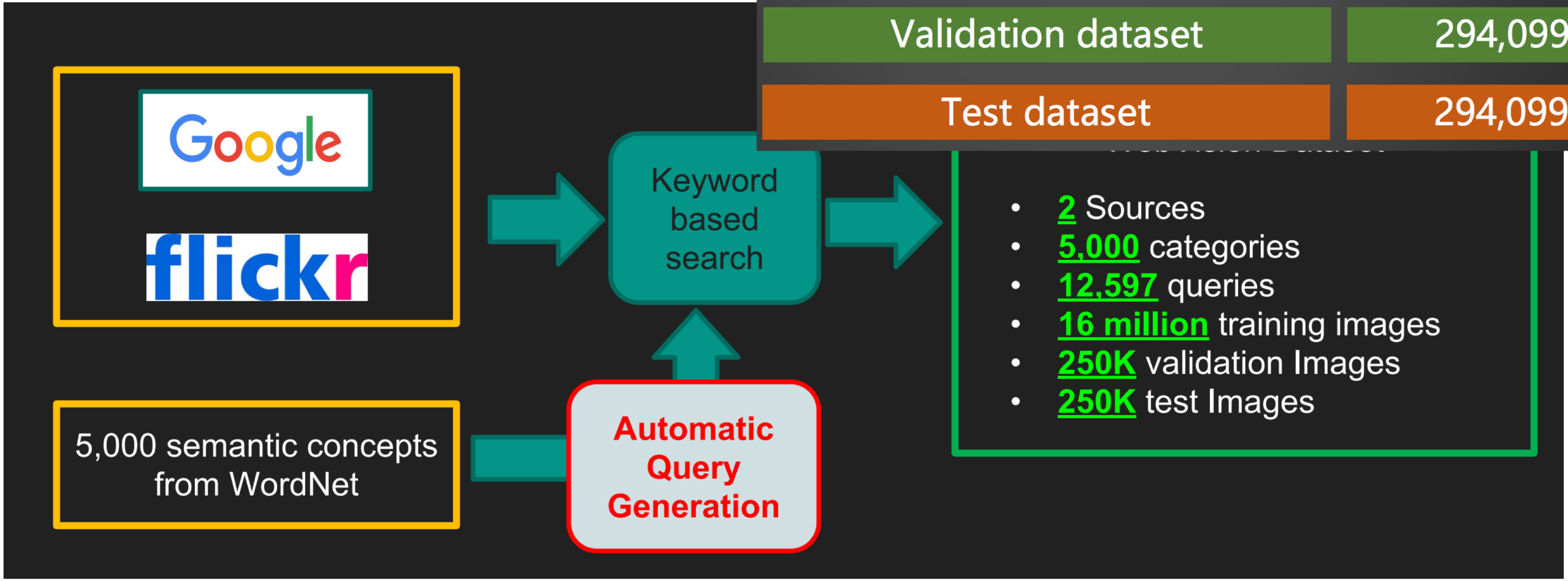
Date: 6/16/2019

Outline

- Challenge analysis
- Our strategy
- Summary

Dataset overview

Name		number of images
Train dataset	Google	8,366,429
	Flickr	7,710,236
Validation dataset		294,099
Test dataset		294,099

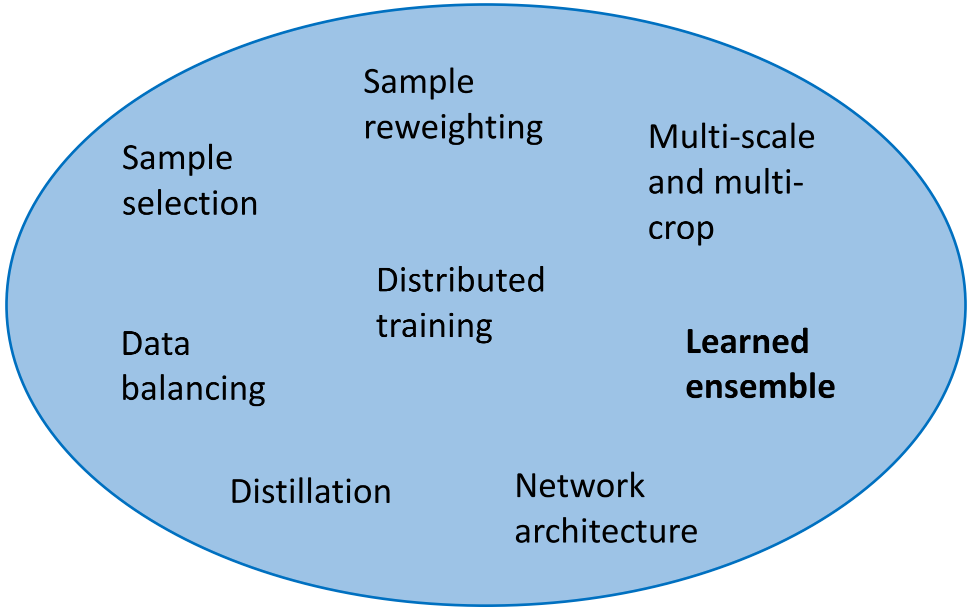


What are the challenges?

- ✓ Large! ~16 million images
- ✓ Imbalanced class distribution
- ✓ Weak/noisy labels:
  - ✓ Incorrect labels: query term as label
  - ✓ Ambiguous labels: apple, corolla
- ✓ High inter-class similarity: banker, psychologist, liar, president, executive…
- ✓ Domain difference between training and testing

Bag of tricks

- ✓Large scale distributed training
- ✓Handle imbalanced class distribution
- ✓Handle noise
  - ✓Sample selection
  - ✓Sample reweighting
- ✓Model architecture
  - ✓ResNet, ResNeXt, SE Block
- ✓Meta information
  - ✓Semantic matching
- ✓Model distillation
- ✓Model ensemble

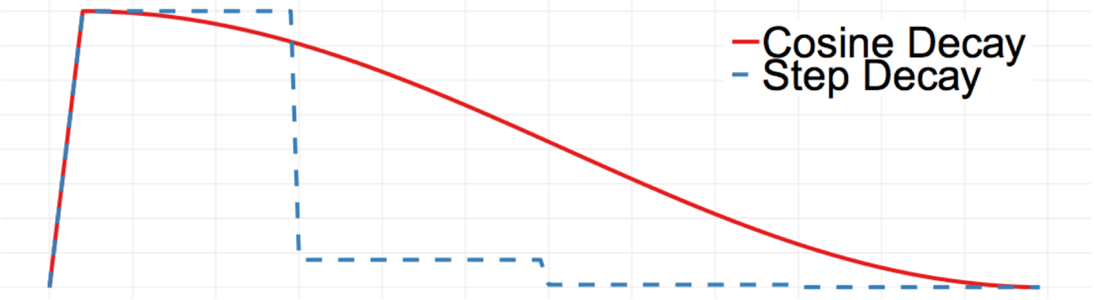


Training strategy

Top-5 accuracy of ResNet-50

Baseline (official)	Our baseline
71.49	73.60

- ✓ Large batch size
- ✓ Warmup + cosine LR
- ✓ Distributed training using Huawei ModelArts
  - ✓ 8 hours per model with 16 GPU's

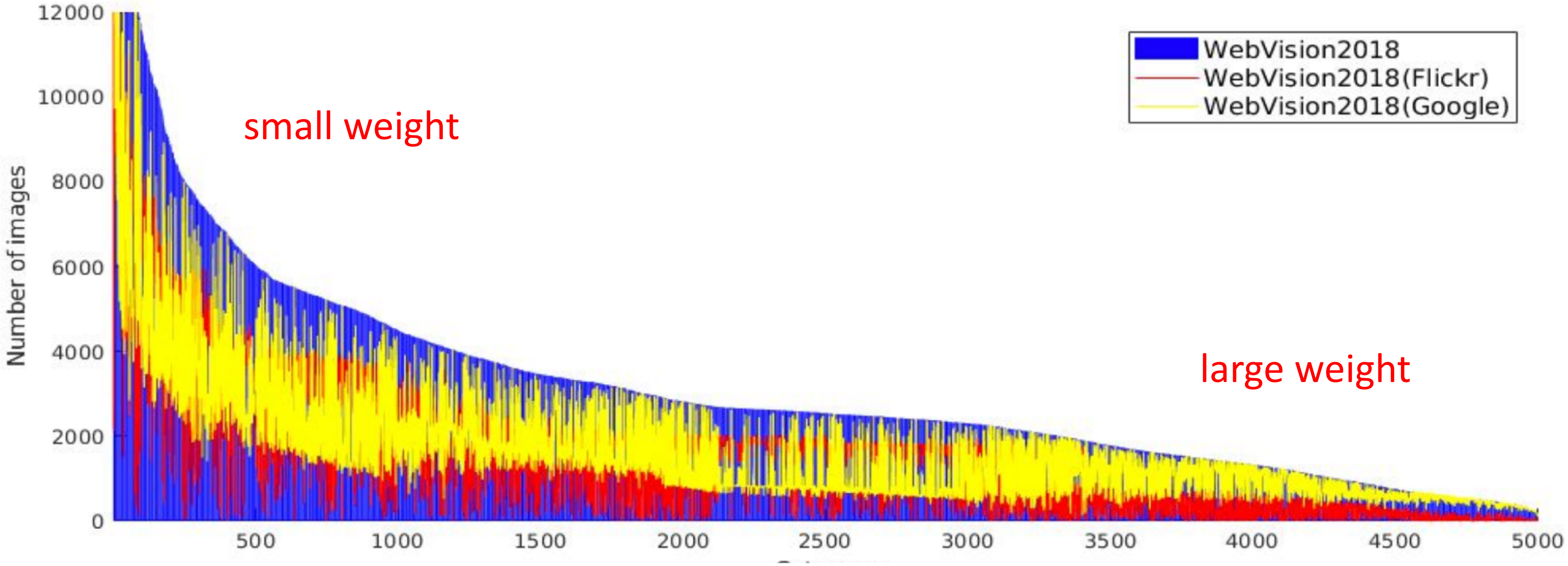


Pretrained models

We offer several pretrained models. **Due to the class imbalance in WebVision, we duplicated the file items in train.txt such that different classes have equal number of training samples. You might want to add similar strategies in imagenet5k.py or modify your own train.txt. Check utils/upsample.py for an example.**

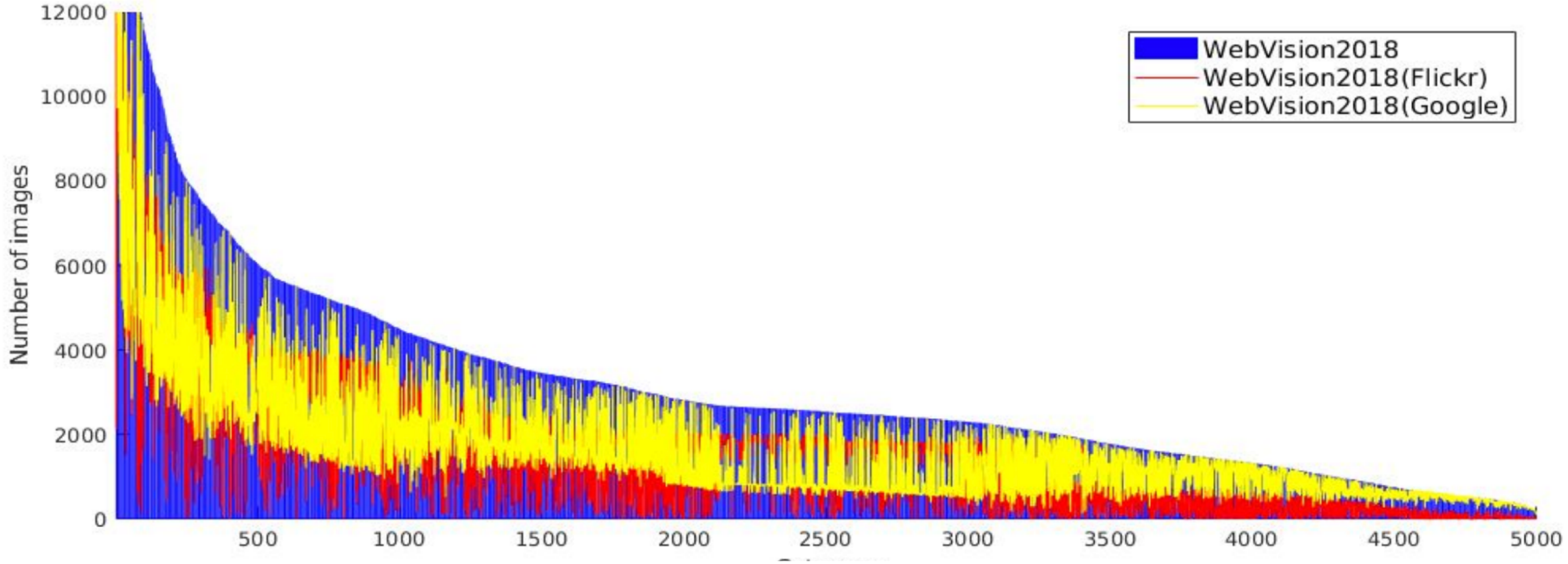
Model	Top1-Val-Error	Top5-Val-Error	Download
ResNet-50 (101 Epoch)	54.28%	30.69%	link
ResNet-50 (205 Epoch)	52.10%	28.51%	link
ResNet-101 (100 Epoch)	52.21%	28.62%	link
ResNet-101 (200 Epoch)	50.12%	26.78%	link
ResNet-101 (300 Epoch)	48.97%	25.74%	link
ResNet-101 (500 Epoch)	48.38%	25.21%	link
ResNeXt-101 (100 Epoch)	50.62%	27.11%	link
ResNet-152 (100 Epoch)	51.23%	27.80%	link
ResNet-152 (200 Epoch)	48.98%	25.75%	link
ResNet-152 (300 Epoch)	48.05%	24.88%	link
ResNet-152 (500 Epoch)	47.31%	24.31%	link
ResNet-152-SE (100 Epoch)	51.61%	28.02%	link

Handle imbalance - reweight class



Baseline	Reweight class
73.6	74.4

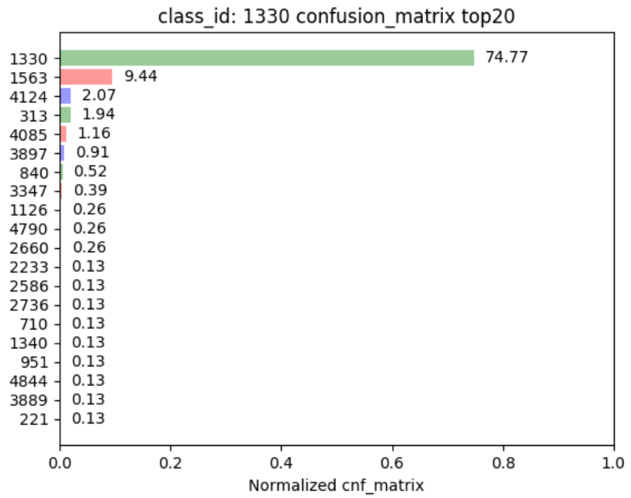
Handle imbalance – top ranked images + oversampling

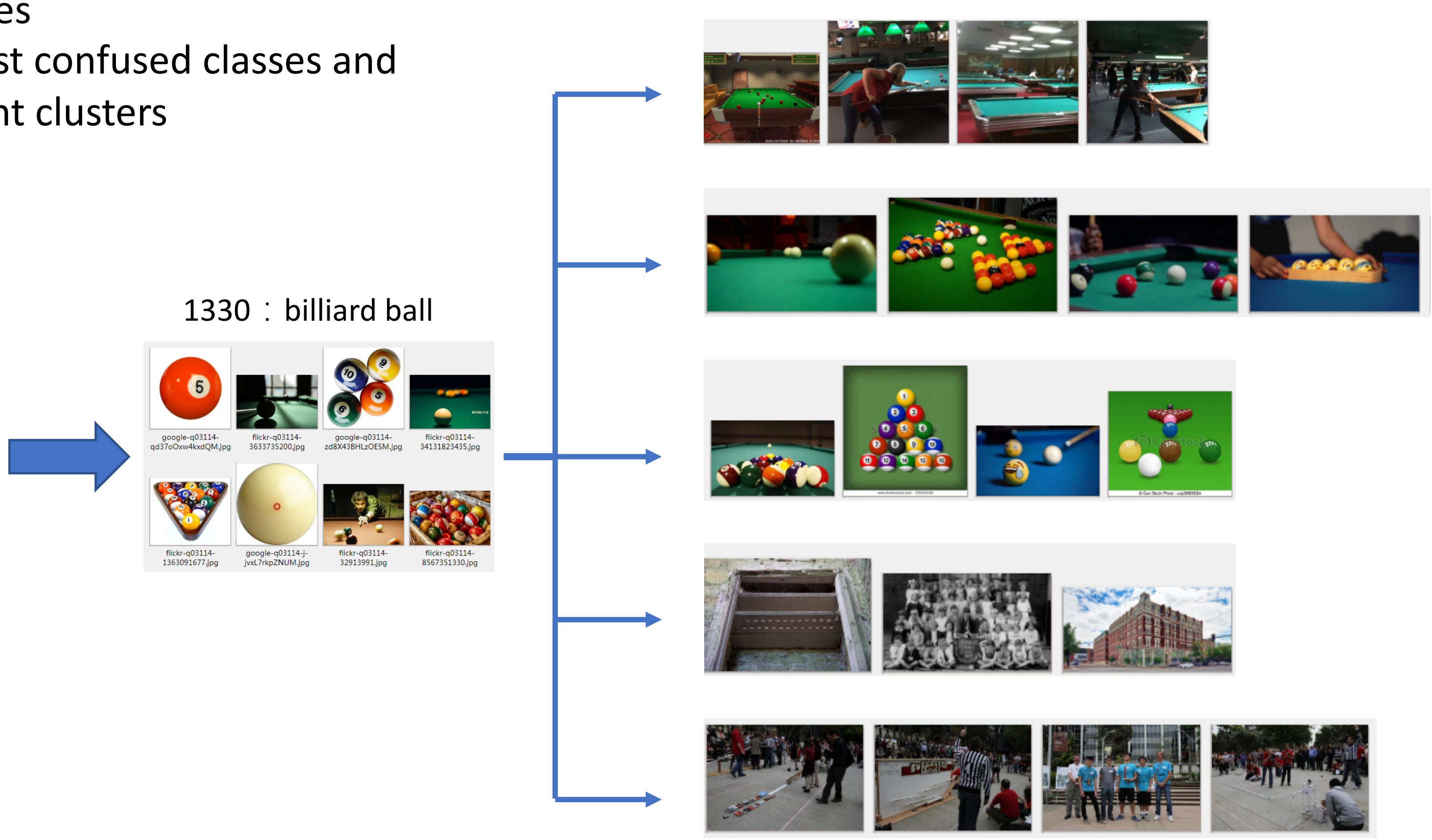


Baseline	Reweight class	Top-3500 ranked images + oversampling
73.6	74.4	75.02

Handle noise - clustering

- ✓ Focus more on confused classes
- ✓ Combine the images from most confused classes and then cluster them into different clusters

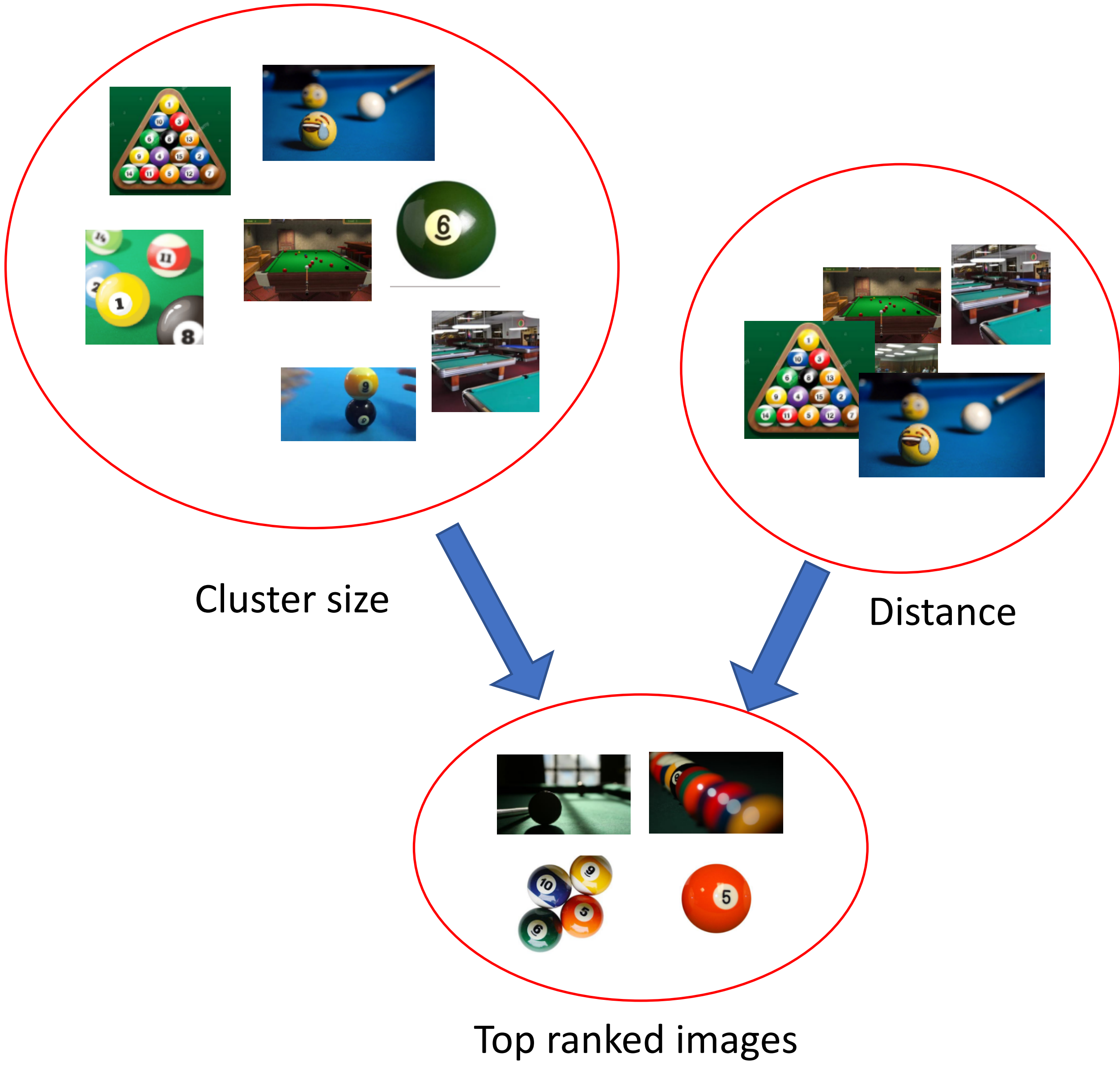




Li et al. Learning from Large-scale Noisy Web Data with Ubiquitous Reweighting for Image Classification. 2019.

How to use the cluster result?

- ✓ Sample selection based on clustering
  - ✓ Choose cluster with more images
  - ✓ Choose cluster with less intra-class variability – compact cluster
- ✓ Sample reweighting based on clustering
  - ✓ Set large weight for large cluster
  - ✓ Set large weight for cluster close to “top ranked images”



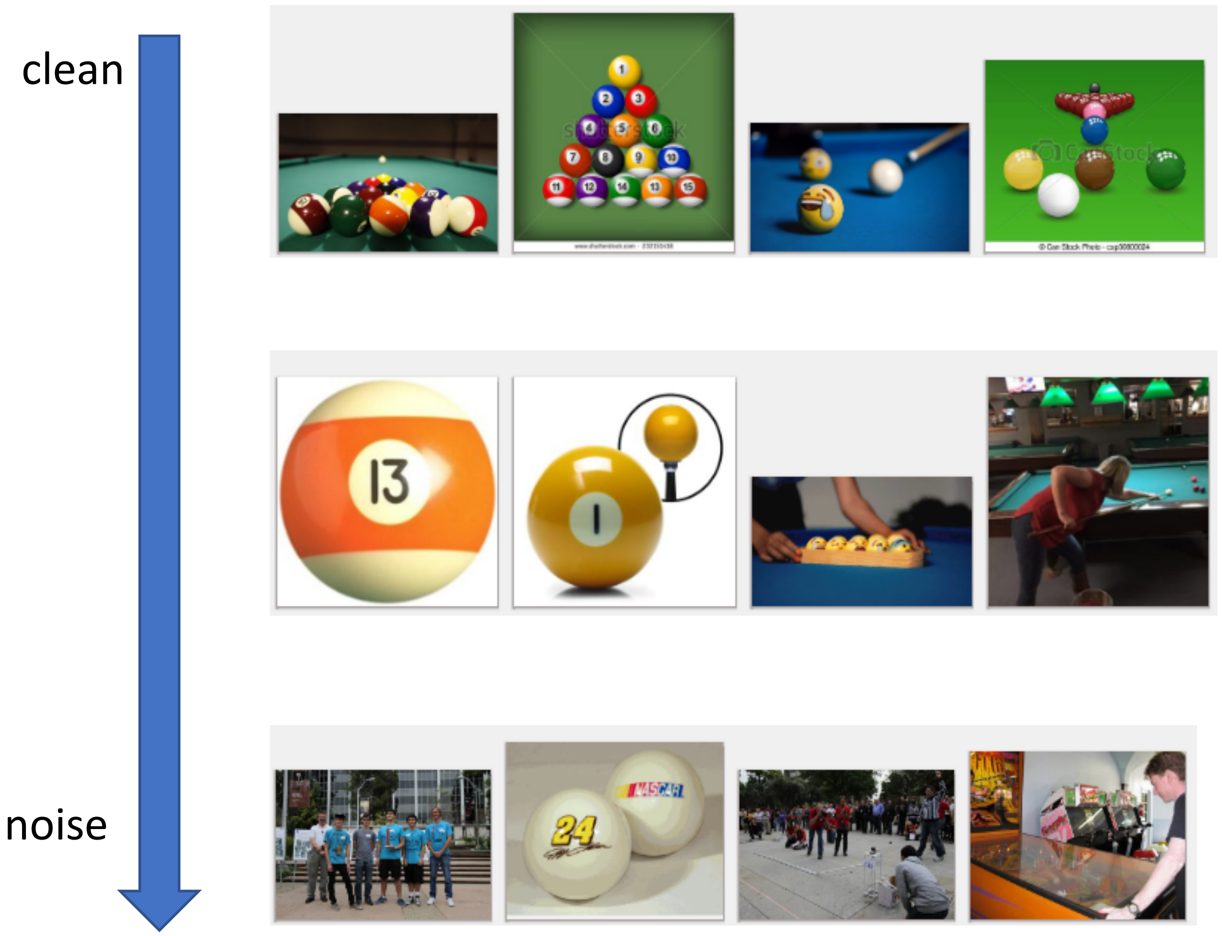
Handle noise - density clustering

- ✓ Assumption: denser -> cleaner
- ✓ For each class, compute the density of each image and then cluster images based on density estimation

Guo et al. CurriculumNet: Weakly Supervised Learning from Large-Scale Web Images. 2018

How to use the cluster result?

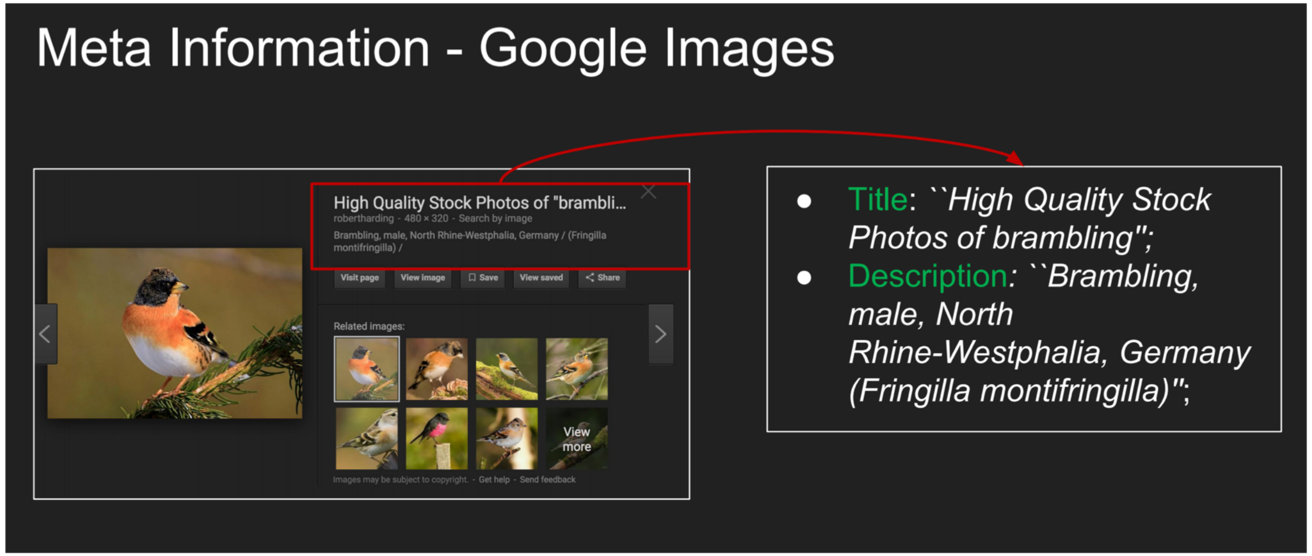
- ✓ Choose the clean cluster for training
- ✓ Use curriculum learning to train model using clean to noisy data in turn
- ✓ Reweight based on cleanness of cluster

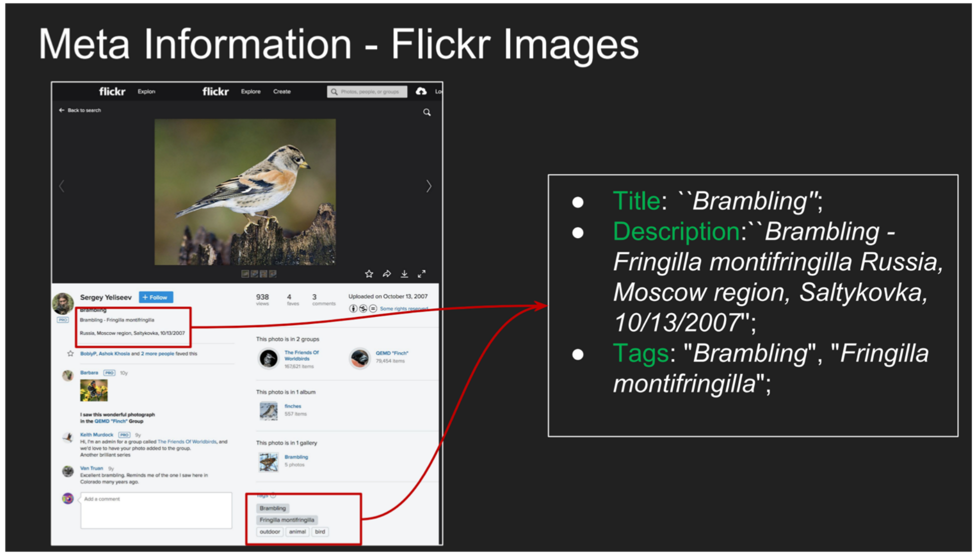


Summary of results based on clustering

Method	Top-5 accuracy (%)
Baseline	73.6
Baseline + reweight class	74.4
K-means + choose large cluster	69.7
K-means + reweight based on cluster size	74.6
K-means + reweight based on distance to top ranked images	74.8
Density clustering + choose clean cluster	70.8
Density clustering + curriculum learning	72.3
Density clustering + reweight cluster	74.8

Leverage meta information





- Sematic match the meta information (descriptions/tags) of image with the description of synset:
  - BERT model: convert descriptions/tags into vectors and compare vectors
  - Keyword matching: match the keywords between image descriptions/tags with synset descriptions

Leverage meta information

{
"description": "There are philosophies as varied as the **flowers** of the field, and some of them weeds and a few of them poisonous weeds. But they none of them create the psychological conditions in which I first saw, or desired to see, the **flower**. - G. K. Chesterton",
"tags": "**flower** quote style philosophy petal stamen chesterton corolla mythoto",

"title": "philosophy"
}

Corolla

N11691046: (botany) the whorl of petals of a flower that collectively from an inner floral envelope or layer of the perianth: “we cultivate the flower for its **corolla**”

{
"description": "2016 Toyota Corolla Ascent Auto",
"title": "New & Used Toyota Corolla Sedan **cars** for sale in Australia ..."
},

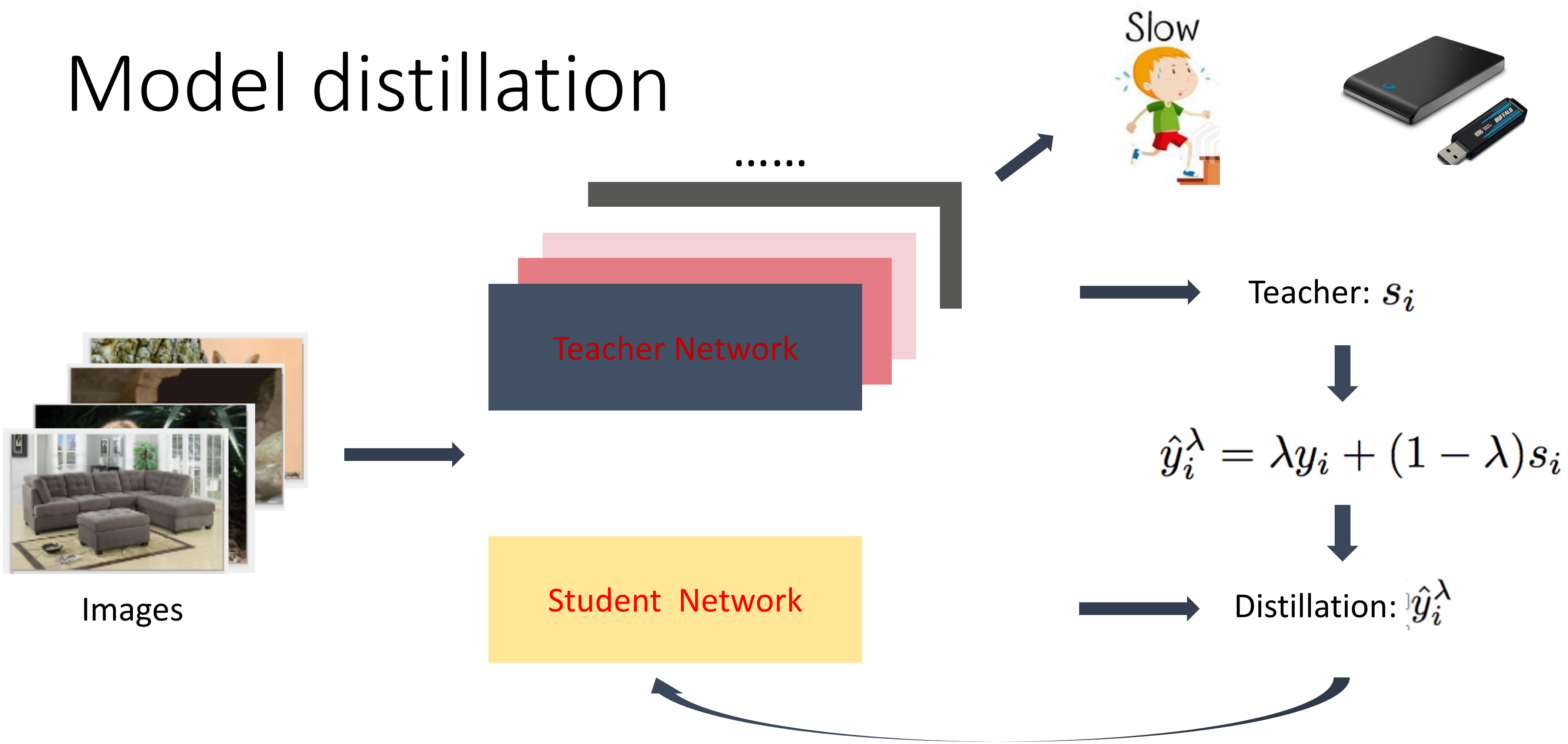
Summary of result based on meta information

Method	Top-5 accuracy (%)
Baseline	73.6
Semantic Matching	65.5
Baseline + Semantic Matching	75.35

Model architecture

Method	Top-5 accuracy (%)
ResNet-50 (baseline)	73.6
SE-ResNet-50	73.12
ResNet-152	76.61
ResNet-200	76.25
ResNeXt-101	75.3

Model distillation



Li et al. Learning from Noisy Labels with Distillation. ICCV. 2017.

Model distillation

Method	Top-5 accuracy (%)
baseline	73.6
Distillation with baseline	74.44
Distillation with top ranked images + oversampling	75.02
Distillation with ensemble of models	77.3

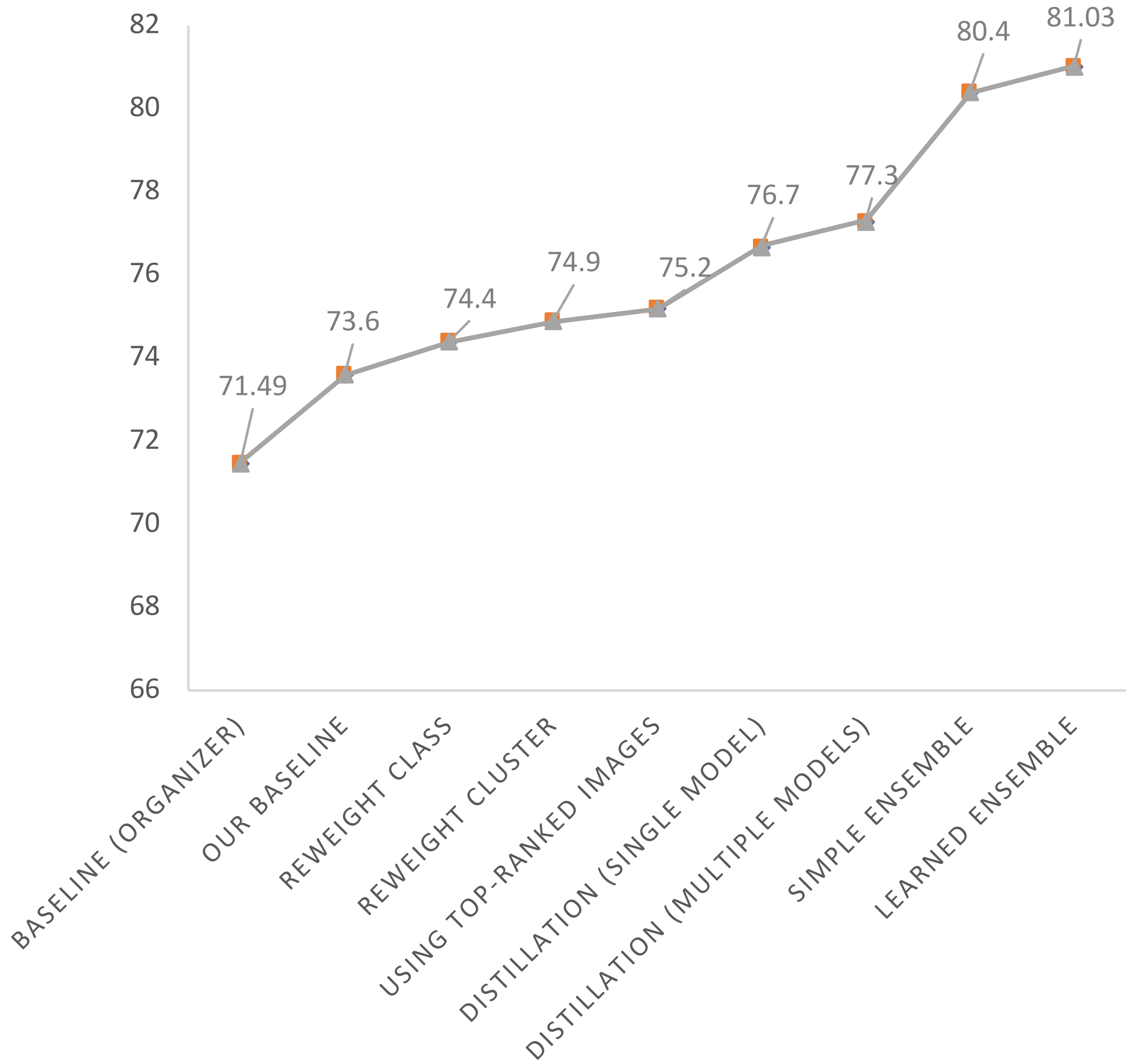
Ensemble strategy

- Majority Voting
- Average combination
- Vote + average logits
- Learn the combination weights

Results of ensemble

Method	Top-5 accuracy (%)
Simple ensemble	80.4
Learned ensemble	81.3

Summary of the results



Challenge Results

Rank	Team name	Top-5 Accuracy (%)
1	**Alibaba-Vision**	**82.54**
2	**BigVideo**	**82.05**
3	**huaweicloud**	**81.15**
4	Y_Y	80.69
5	PCI	77.92

Take-home messages

- ✓Good model architecture generally leads to better performance
- ✓Semantic information is useful especially when combined with visual information
- ✓Model distillation originally proposed for model compression works quite well for learning from web data
- ✓Top-ranked images are more useful
- ✓ Ensemble always helps, and learned ensemble is even better

Thanks ☺