Dataset Culling: Towards Efficient Training of Distillation-based Domain Specific Models

K. Yoshioka(1)(2), E. Lee(2), S. Wong(2), M. Horowitz(2)
(1) Toshiba
(2) Stanford University

IEEE ICIP 2019 Sept. 25
Introduction

• Deep Learning based object detection has excellent accuracy.
  • e.g. Vision for security, infrastructure, transportation..

• Cost?
  • Requires many GPU-hours, difficult to scale.
  • Has accuracy-cost tradeoff.

101-layer Resnet: Imagenet accuracy 78%
10-layer Resnet: Imagenet accuracy 60%

• How can we break this tradeoff?
Introduction: Domain Specific Models

- Training compact domain specific models (DSMs) [1,2]
  - DSMs: a specialized model for specific env. {conference room, your house, your office, etc.}
  - Cuts down computation cost 5-20x

![General dataset](http://cocodataset.org/)

Introduction: What is Distillation?

- Teacher model teaches the small student model to learn
  - Works without human interference

**Diagram: Distillation Process**

1. **Domain data**
2. **Teacher model (large, general)**
   - Provides "answers"
3. **Domain Specific Model (Small, specialized)**
4. **Train model**
Introduction: The Problem

- Can gather lots of training data easily..
  - A day’s worth of surveillance data = 86,400 images @ 1FPS
- Training 86,400 images require over 100 GPU-hours (Nvidia K80 on AWS) to train.
  - Unable to scale to deploying thousands of cameras
- Reducing the DSM training cost has not been explored.
Dataset Culling
Basic Idea of Dataset Culling

- Reduces the dataset size \(300x\)
- Culls only “Easy” data; model accuracy is not harmed

Total training time: 104 \(\rightarrow\) 2.2 GPU-hours

47x improvement 😊
What is good training data?

• “Difficult” data which the model makes a lot of mistakes.
  • No backprop is done if the model can perfectly predict.
    → Does not contribute to training.
  • Comparing teacher-student predictions are costly..

• Can we assess from student predictions only?
How can we pick good training data?

- Quantify good data by proposed “confidence loss”
- Assesses the difficulty of prediction from the output probability

![Image showing detection confidence and loss computation](image_url)

**Car=0.49**  
**Car=0.59**  
**Car=1.0**  
**Car=0.79**

**Image Conf. Loss: 3.79**
Dataset culling pipeline

- First, cull dataset using **only** the student model
  - Culls out majority of the data first (50x).
  - Cheap; does not require costly teacher inference.

```
1) Cull by confidence loss $L_{conf}$
   - Compute student prediction
   - Compute confidence loss ($L_{conf}$)
   - Pick $n$ data with largest $L_{conf}$
```
Dataset culling pipeline

- Then, conduct a secondary culling using both teacher-student predictions.
  - Directly determine errors the student makes.
  - Data is culled up to 300x by the pipeline.
Experiments
Experiment setups

• Models pretrained on MS-COCO:
  • Student: Resnet-18 based Faster-RCNN
  • Teacher: Resnet-101 based Faster-RCNN

• Dataset: 8 custom videos acquired from Youtube.
  • Train: first 24-hours
  • Validation: Subsequent 6-hours
  • Utilize teacher output as ground-truths
Qualitative results

<table>
<thead>
<tr>
<th>Model</th>
<th>mAP</th>
<th>comp</th>
</tr>
</thead>
<tbody>
<tr>
<td>RawStudent</td>
<td>78.2</td>
<td>28G</td>
</tr>
<tr>
<td>TrainStudent</td>
<td>90.2</td>
<td>28G</td>
</tr>
<tr>
<td>TrainStudent+optResolution</td>
<td>89.6</td>
<td>7G</td>
</tr>
<tr>
<td>Teacher</td>
<td>93.2</td>
<td>128G</td>
</tr>
<tr>
<td>Oracle</td>
<td>80.7</td>
<td>18G</td>
</tr>
</tbody>
</table>
Quantitative Results

- Can cull the dataset size to 300x, without accuracy drops or even with improvements.

<table>
<thead>
<tr>
<th>Culled dataset size</th>
<th>64</th>
<th>128</th>
<th>256</th>
<th>Full (86,400)</th>
<th>No Training</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Accuracy [mAP]</td>
<td>85.56</td>
<td>88.3</td>
<td>89.3</td>
<td>88.5</td>
<td>58.6</td>
</tr>
<tr>
<td></td>
<td>(-3.0%)</td>
<td>(-0.3%)</td>
<td>(+0.8%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total train time [hours]</td>
<td>1.9 (54x)</td>
<td>2.0 (50x)</td>
<td>2.2 (47x)</td>
<td>104</td>
<td>-</td>
</tr>
<tr>
<td>Student Prediction [hours]</td>
<td>1.54</td>
<td>1.54</td>
<td>1.54</td>
<td>0</td>
<td>-</td>
</tr>
</tbody>
</table>
Conclusions

• While DSMs can reduce the inference cost, training them can take many GPU-hours.

• We proposed Dataset Culling, which reduces the DSM training cost by 47x.
  • Only easy to predict data are culled to minimize the accuracy drop.
  • Evaluated on our long-duration dataset, we saw little to no accuracy penalty even with culling.

Codes and dataset available:
https://github.com/kentaroy47/DatasetCulling
Ablation study

• Entropy implements the loss function for active learning.

• Using teacher-student comparisons achieve best accuracy (Precision)

• Our dataset culling pipeline with Confidence + Precision has the best tradeoff of accuracy and training time.

<table>
<thead>
<tr>
<th>Filtering strategy</th>
<th>Intermittent Samp.</th>
<th>Entropy[9]</th>
<th>Confidence</th>
<th>Precision</th>
<th>Confidence + Precision</th>
<th>Full dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>mAP</td>
<td>0.731</td>
<td>0.866</td>
<td>0.911</td>
<td>0.954</td>
<td>0.948</td>
<td>0.958</td>
</tr>
<tr>
<td>GPU hours</td>
<td>0.15</td>
<td>1.7</td>
<td>1.7</td>
<td>8.0</td>
<td>2.0</td>
<td>104</td>
</tr>
</tbody>
</table>

© 2019 Kentaro Yoshioka