The Fault in Our Data Stars: **Studying Mitigation Techniques against Faulty Training Data** in Machine Learning Applications

*Abraham Chan, Arpan Gujarati, Karthik Pattabiraman, Sathish Gopalakrishnan*

*The University of British Columbia*
Training Data

<table>
<thead>
<tr>
<th>Image</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td>Normal</td>
</tr>
<tr>
<td><img src="image2.png" alt="Image" /></td>
<td>Pneumonia</td>
</tr>
</tbody>
</table>
Modern ML Applications

Dataset: Pneumonia

Prediction: Normal

Model Accuracy: 90%
Training Data Faults

- **Mislabelling**: Image of pneumonia labeled as normal.
- **Removal**: Image labeled as pneumonia but marked with an X to indicate removal.
- **Repetition**: Two images labeled as pneumonia.
Random Mislabelling

Dataset: Pneumonia

Actual: **Normal**
Prediction: **Pneumonia**
Model Accuracy: **55%**
Original Accuracy: **90%**
20% of ChestX-ray mislabelled [Tang et al, 2021]

33% of the popular Udacity Dataset2 mislabelled or missing labels [Dwyer, 2020]
How can we mitigate training data faults?

(From the P.O.V of a practitioner)
Selection Criteria

- Code Available?
- Architecture agnostic?
- Artificial noise?
- Not pre-trained?
- Standalone?
Techniques against Mislabeled Data

1. Loss Correction (LC)
2. Knowledge Distillation (KD)
3. Robust Loss (RL)
4. Label Smoothing (LS)
5. Ensemble Learning (Ens)

Our Contribution:
How do we choose a technique?
Techniques against Mislabeled Data

1. Loss Correction (LC)
2. Knowledge Distillation (KD)
3. Robust Loss (RL)
4. Label Smoothing (LS)
5. Ensemble Learning (Ens)

More Practitioner Effort

Less Practitioner Effort
Loss Correction (LC)

Techniques:

- LC
- KD
- RL
- LS
- Ens

Main Model

Faulty Training Dataset

Label Correction

Inner Model

Clean

Subset
Self Knowledge Distillation (KD)

- Teacher = Student = (i.e., ResNet50)
Robust Loss (RL)

ML Model

Class #1: 0.775
Class #2: 0.116
Class #9: 0.030
Class #10: 0.040

Predicted Probabilities

Loss Function

Actual Probabilities

Class #1: 1
Class #2: 0
Class #9: 0
Class #10: 0

Techniques:

LC  KD  RL  LS  Ens
Robust Loss (RL)

Techniques: LC KD RL LS Ens

ML Model

Predicted Probabilities

Class #1: 0.775
Class #2: 0.116
Class #9: 0.030
Class #10: 0.040

Cross Entropy

Actual Probabilities

Class #1: 1
Class #2: 0
Class #9: 0
Class #10: 0
Robust Loss (RL)

ML Model

Class #1 0.775
Class #2 0.116
Class #9 0.030
Class #10 0.040

Predicted Probabilities

Normalized Cross Entropy + Reverse Cross Entropy

1 0
0 0
0 0

Actual Probabilities

Class #1
Class #2
Class #9
Class #10

Found to be more robust [Ma et al., 2020]

Techniques: LC KD RL LS Ens
Label Smoothing (LS)

Without LS

Label: [0, 0, 1, 0, 0]

With LS

Label: [0.05, 0.1, 0.7, 0.1, 0.05]

Smaller Gradient
Understanding the Resilience of Neural Network Ensembles against Faulty Training Data
Our Prior Work: [QRS’21]
Methodology

Techniques:

- LC
- KD
- RL
- LS
- Ens

Mitigation Technique(s):

- Baseline
- Ens
- LS
- RL
- KD
- LC

Accuracy Delta (AD)

Golden Model

Training Data

Faulty Model
# Neural Networks

<table>
<thead>
<tr>
<th>ML Model Name</th>
<th>Depth (# of Layers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConvNet</td>
<td>Shallow</td>
</tr>
<tr>
<td>DeconvNet</td>
<td>Shallow</td>
</tr>
<tr>
<td>MobileNet</td>
<td>Deep</td>
</tr>
<tr>
<td>ResNet18</td>
<td>Deep</td>
</tr>
<tr>
<td>ResNet50</td>
<td>Deep</td>
</tr>
<tr>
<td>VGG11</td>
<td>Deep</td>
</tr>
<tr>
<td>VGG16</td>
<td>Deep</td>
</tr>
</tbody>
</table>
Evaluation Datasets

CIFAR-10
Object Detection

GTSRB
Self-Driving Cars

Pneumonia
Medical Diagnosis

Safety-Critical Applications
Reliability Metric: Accuracy Delta (AD)

Model trained with golden data

- Test Image 1: ✔
- Test Image 2: ✔
- Test Image 3: ✔
- Test Image 4: ❌

Model trained with faulty data

- Test Image 1: ✔
- Test Image 2: ❌
- Test Image 3: ❌

Accuracy Delta (AD) = 2 / 3 = 67% in this case
AD Across:

Models

Fault Types

Datasets

- GTSRB, ResNet50, Mislabelling
- GTSRB, VGG16, Mislabelling

KD is not effective here

KD is effective here

Higher AD is worse
Ensembles are effective across models

Higher AD is worse

GTSRB, ResNet50, Mislabelling

GTSRB, VGG16, Mislabelling
AD Across: Models  Fault Types  Datasets

Higher AD is worse

GTSRB, ResNet50, Mislabelling  GTSRB, ResNet50, Removal

LS is also effective across fault types
Ensembles are also effective across fault types

Higher AD is worse

GTSRB, ResNet50, Mislabelling  GTSRB, ResNet50, Removal
Finding: Ensemble is generally effective, followed by LS

Higher AD is worse

GTSRB
Self-Driving Cars

Pneumonia
Medical Diagnosis
Takeaways

• **Ensembles** performed best overall but **Label Smoothing** surprisingly effective (second place)

• Dataset size did not have an impact on **Loss Correction** (but works well for datasets with fewer classes)

• **Knowledge Distillation** and **Robust Loss** performed well only at low fault amounts
Summary

• **Problem:** Choose a mitigation technique against faulty training data

• **Approach:** Evaluate techniques on 7 models across 3 datasets

• **Results:**
  – **Ensembles** effective across all configurations
  – **Label smoothing** is second in effectiveness, with less overhead

**Email:** abrahamic@ece.ubc.ca