Learning from Noisy Labels for Earth Observation

Prof. Dr. Begüm Demir

Remote Sensing Image Analysis (RSiM) Group, Faculty of EECS, TU Berlin
Big Data Analytics for Earth Observation (BigEarth) Group, BIFOLD
BIFOLD: Berlin Institute for the Foundations of Learning and Data

- Declarative ML in selected applications
- ML in heterogeneous datasets and data streams
- Scalable processing of heterogeneous, geo-distributed data streams
- Near real-time processing of millions of data sources

2014-2022

- Theoretical and algorithmic foundations of ML
- New scientific and technical ML applications
- Usability of ML methods for business and science
- New interdisciplinary research in science and medicine

2018-2022
Introduction

➢ The two tasks common to most of the EO applications are:

- land-use land-cover (LULC) maps generation and updating

- automatic scene understanding based on the LULC classes

Realization of these tasks requires a large amount of annotated training samples. Collecting annotations is costly in terms of human time/effort and needs expertise.
Thematic Products as Labelling Sources

Use the publicly available **thematic products** (digital maps).

- Thematic products can be used to generate large scale training sets with zero-labeling-cost, covering wide areas, such as:
  - **Global scale**, e.g., Global Land Cover, the ESA CCI-LC product;
  - **Continental scale**, e.g., Corine Land Cover maps at European level;
  - **National scale**, e.g., American National Land Cover Dataset, DFD Land Use and Land Cover Product for Germany.

Thematic Products as Labelling Sources

➢ Thematic maps can produce noisy labels due to:

- errors in the map due to different the approaches used to generate the map;
- changes in land-use/land-cover after the construction of the maps;
- errors due to the misalignment between the map and the satellite image.

➢ Two types of label noise can be present in a training image:

- missing labels;
- wrong labels.

Training sets with noisy labels can distort the learning process. Methods that are robust to the label noise are required.
Learning from Noisy Labels

➢ There are several label-noise robust methods, relying on:
  ▪ estimation of a noise transition matrix;
  ▪ noise robust loss functions;
  ▪ building class prototypes;
  ▪ student-teacher networks;
  ▪ data augmentation techniques.

Most of these works target the wrong labels only and thus not directly suitable for multi-label scene analysis problems in RS.

Multi-Label Classification in Remote Sensing

➢ Automatic scene understanding based on the LULC classes

**Solution:** We have developed a robust consensual collaborative learning method (RCML).
RCML: Robust Collaborative Multi-Label Learning Method


RCML Modules

The **Discrepancy Module** aims at forcing the two networks to learn diverse features, while achieving consistent predictions.

It includes: 1) Disparity loss ($L_D$); and 2) Consistency loss ($L_C$).

The **Group Lasso Module**:

- identify potentially noisy labels by using the predictions of the two networks;
- identify the type of label noise by computing a sample-wise ranking loss as:

**Lasso**($x_i$) = $\alpha \sum_{\hat{c}=a+1}^{m} \begin{cases} 0 & \text{if } \hat{c} = c+1 \\ \varepsilon \quad \text{otherwise} \end{cases} + \beta \sum_{c=b+1}^{m} f_c(x_i) - \sum_{\hat{c}=a+1}^{m} f_{\hat{c}}(x_i) + 1$

```
<table>
<thead>
<tr>
<th>Missing label</th>
<th>Wrong label</th>
<th>$f_c$</th>
<th>$f_{\hat{c}}$</th>
<th>$E_{c,\hat{c}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>×</td>
<td>×</td>
<td>0</td>
<td>0</td>
<td>+1</td>
</tr>
<tr>
<td>×</td>
<td>×</td>
<td>0</td>
<td>0</td>
<td>+1</td>
</tr>
<tr>
<td>×</td>
<td>×</td>
<td>1</td>
<td>1</td>
<td>+1</td>
</tr>
<tr>
<td>×</td>
<td>×</td>
<td>1</td>
<td>1</td>
<td>+1</td>
</tr>
</tbody>
</table>
```
The Swap Module:
- exchanges the ranking information between the networks;
- takes the Binary Cross Entropy and ranking losses into consideration to eliminate the detected noisy samples from back-propagation.

\[ B_i^f = L_f(x_i) + \gamma \text{Lasso}_f(x_i) \]

\[ \text{Loss}^b_f = \frac{1}{R} \sum_{i=1}^{R} L_f(x_i) + \lambda_1 L_c + \lambda_2 L_D \]
In the experiments we used our BigEarthNet dataset that:

- consists of 590,326 Sentinel-1&2 images.
- opens up promising directions to support studies for the analysis of large-scale EO data archives.


The ResNet50 architecture was considered as backbone for our RCML.

We report the results of training obtained after 100 epochs.

The model training and further experiments were conducted on a Tesla V100 GPU with 32 GB RAM.

In our experiments, the ratio of the introduced noise was varied from 10% to 50%.

The results of the experiments were provided in terms of: 1) mean average precision (mAP); and 2) F1 score.

We compared our proposed method with four baseline methods:


RCML: Results on BigEarthNet

Urban fabric

Arable lands

Pasture

Moors, heathland and sclerophyllous vegetation

Inland wetlands

Marine Waters

FL: focal loss
SAT: self-adaptive training
ELR: early-learning regularization
BCE: standard binary cross-entropy
## RCML: Results on BigEarthNet

<table>
<thead>
<tr>
<th>Noise Rate</th>
<th>$F_1$ (%)</th>
<th>mAP (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FL</td>
<td>SAT</td>
</tr>
<tr>
<td>0%</td>
<td>64.0</td>
<td>66.0</td>
</tr>
<tr>
<td>10%</td>
<td>62.4</td>
<td>65.0</td>
</tr>
<tr>
<td>20%</td>
<td>55.9</td>
<td>62.4</td>
</tr>
<tr>
<td>30%</td>
<td>48.5</td>
<td>56.2</td>
</tr>
<tr>
<td>40%</td>
<td>39.0</td>
<td>48.3</td>
</tr>
<tr>
<td>50%</td>
<td>30.7</td>
<td>34.0</td>
</tr>
</tbody>
</table>

- **FL**: focal loss
- **SAT**: self-adaptive training
- **ELR**: early-learning regularization
- **BCE**: standard binary cross-entropy
Concluding Remarks

➢ Developing efficient techniques for handling label noise is becoming more and more important.

➢ RCML is promising since it:
  ▪ is able to automatically identify two different types of noise without making any prior assumption;
  ▪ is architecture-independent, and thus can be used within different network architectures;
  ▪ is applicable in different EO applications (e.g., large scale image retrieval, etc.).
The codes developed and maintained at our group are publicly available:

https://rsim.berlin/software

Follow us in twitter: @BigEarthERC