Heterogeneous Model Reuse via Optimizing Multiparty Multiclass Margin

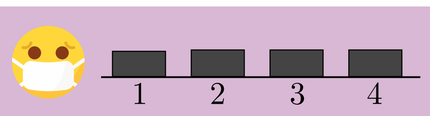
Xi-Zhu Wu[1], Song Liu[2 3], Zhi-Hua Zhou[1]

[1]LAMDA Group, Nanjing University, China

[2]University of Bristol, [3]The Alan Turing Institute, United Kingdom

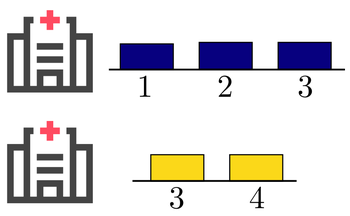
Problem setting

- Problem: Multiparty multiclass classification
- Example: Flu detection



Global problem: to detect all 4 flu types in the US

But, the types of flu diverse geographically, the distribution of patients' records collected by a hospital in California is different from Florida. Good local models are built:



Local model in California detects {1,2,3} types

Local model in Florida detects {3,4} types

The patients' records are confidential. Can we smartly reuse the local models to learn the global problem, instead of building a model on merged local datasets?

Multiple models trained separately with different subsets of the label space

One model works well on the full label space

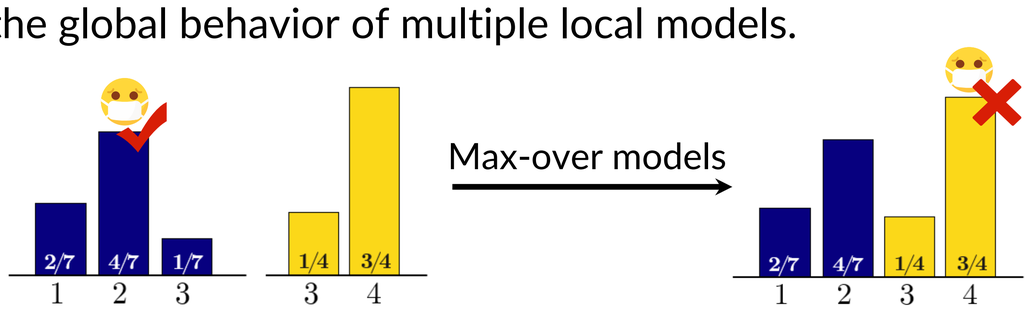
Contribution

Q: How to measure the global behavior of an ensemble?
A: Multiparty multiclass (MPMC) margin.

Q: How to optimize the global behavior?
A: The HMR method, which maximizes MPMC-margin.

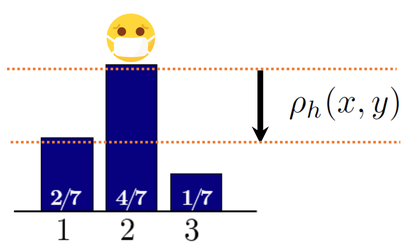
Measure the behavior of multiple models

Simply max-over outputs of multiple local models can be wrong. We propose multiparty multiclass margin to measure the global behavior of multiple local models.



(Single-party) multiclass margin:

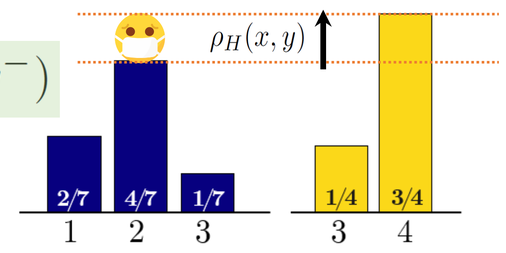
$$\rho_h(x,y) = h(x,y) - \max_{y' \neq y} h(x,y')$$



Multi-party multiclass margin:

$$\rho_H(x,y) = h_{i^+}(x,y) - h_{i^-}(x,y^-)$$

most correct party i^+ vs. most incorrect party i^-



We want multiparty multiclass (MPMC) margin to be positive!

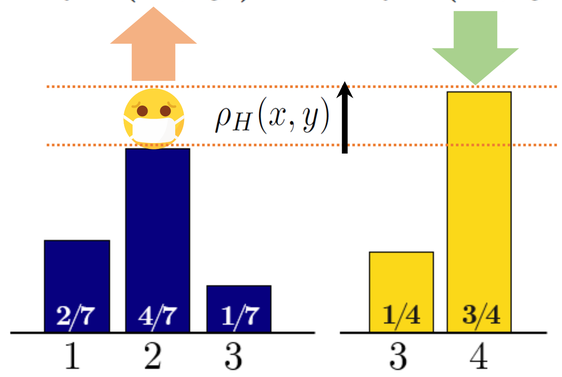
Mail: wuxz@lamda.nju.edu.cn | wuxz.gm@gmail.com

Code: https://github.com/YuriWu/HMR

Maximize MPMC-margin

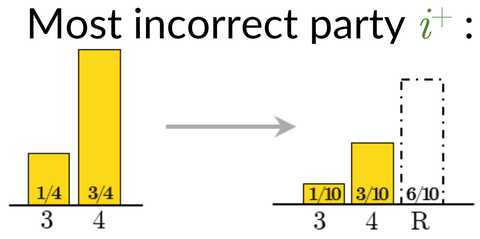
$$\rho_H(x,y) = h_{i^+}(x,y) - h_{i^-}(x,y^-)$$

Increase the first term & decrease the second



Most correct party i^+ : Increase $h_{i^+}(x,y)$ by adding (x,y) to local training data

Most incorrect party i^+ : decrease $h_{i^-}(x,y^-)$



by increase $h_{i^-}(x,R)$

by adding (x,R) to training data

Detailed algorithm

Algorithm 1 HMR

input:
Parties $1,2,\cdots,n$, each owns a local dataset S_i and a local model h_i. Example communication budget N.

output:
Calibrated local models $h_1,\cdots,h_n$.

procedure:

```
Each party broadcasts its local model to others.
Inner iteration counter T = 0
while T < N do
    Sample a party i according to |S_i| / Σ_{i=1}^n |S_i|.
    Party i randomly selects an example (x, y) ∈ S_i.
    Party i computes MPMC-margin ρ_H(x, y) and records the party i^+, i^- and maximum incorrect class y^- as in (8).
    if ρ_H(x, y) ≤ 0 then
        Party i sends (x, y, y^-) to i^+ and i^-.
        Party i^+ calibrates h_{i^+} with (x, y, y^-).
        Party i^- calibrates h_{i^-} with (x, y, y^-).
        Party i^+ and i^- broadcast their updated model.
        if i^+ ≠ i or i^- ≠ i then
            T = T + 1.
        end if
    end if
end while
```

An iterative method exchanges T examples and maximizes MPMC-margin on "unobserved" merged global dataset.

$$\rho_H(x,y) = h_{i^+}(x,y) - h_{i^-}(x,y^-)$$

$$i^+ = \arg\max_i h_i(x,y), \text{ where } y \in \mathcal{Y}_i, \quad (i^-,y^-) = \arg\max_{j,y'} h_j(x,y'), \text{ where } y' \in \mathcal{Y}_j \backslash \{y\}. \tag{8}$$

check MPMC-margin locally

send out one example if non-positive

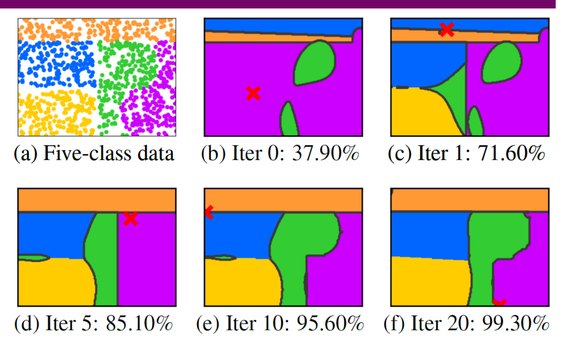
update local models

Privacy issue: If T is small enough, most of the private local data will be protected. Less than 1% of global data are shared in experiments.

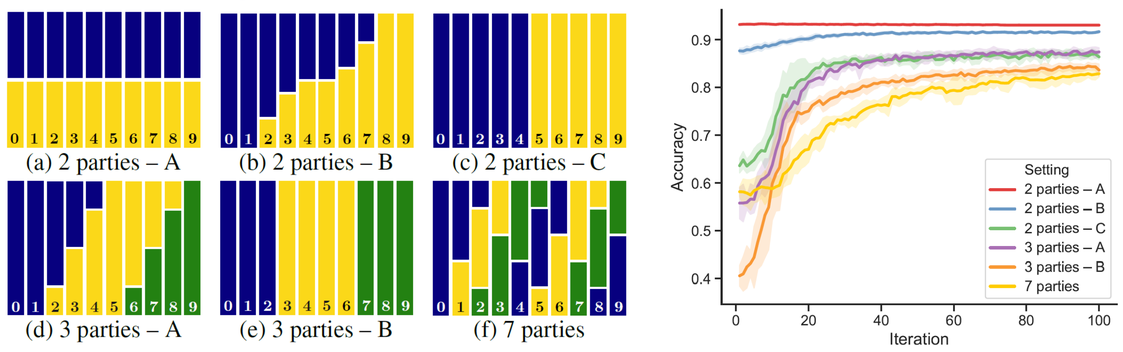
Experiments

Toy example on LR/SVM/GBDT

- Heterogeneous learning models
  - LR: blue, yellow
  - SVM: green, magenta
  - GBDT: magenta, orange
- Exchanged 20 examples
- Nearly perfect performance



Benchmarking on fashion-MNIST on various data partitions



Multi-lingual handwriting recognition

- 6 different structured CNNs trained locally on hiragana, katakana, kanji, devanagari, hangul and English letters
- 1600+ classes, 94.32% global accuracy
- Non-private single model: 96.38%
- Only exchanged 300 out of 420k examples (about 0.07% data)