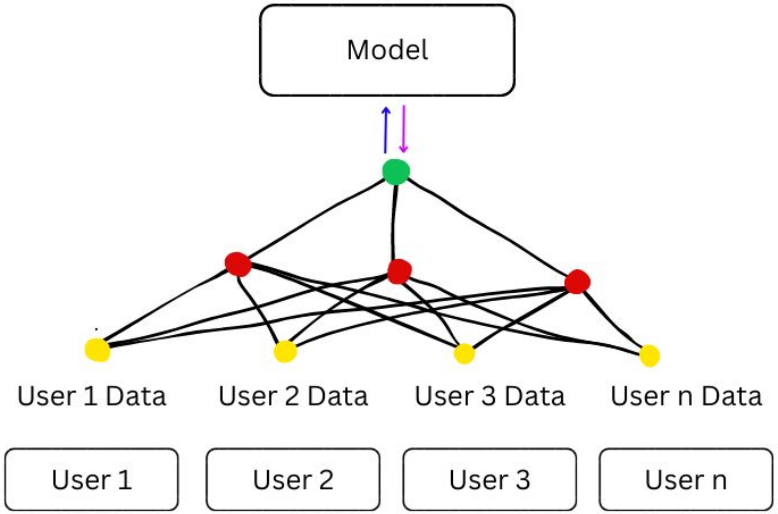
Addressing Statistical Heterogeneity in Federated Learning For Sea Ship Datasets

Serena Lin[1], Emon Dey[2], Anuradha Ravi[2], Nirmalya Roy[2]
[1]Department of Electrical and Computer Engineering, Northeastern University
[2]Department of Information Systems, University of Maryland in Baltimore County

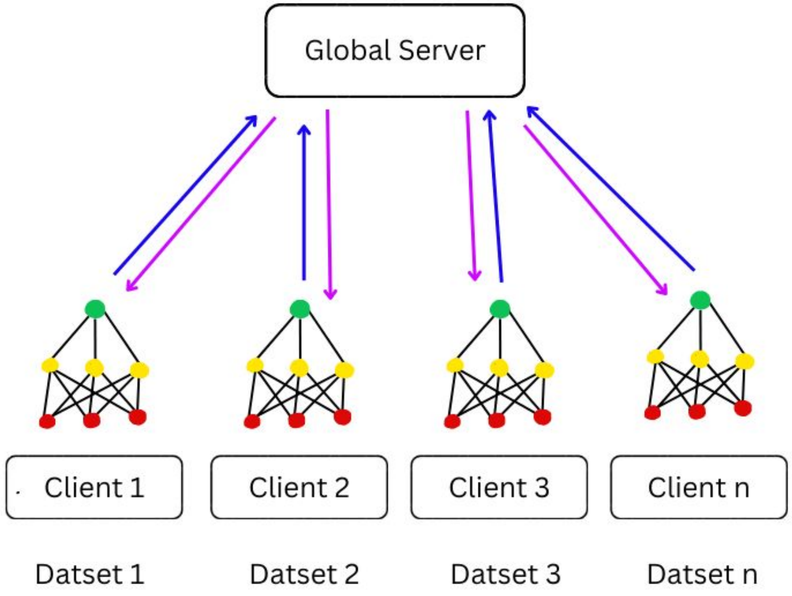
UMBC

We propose a **method** to *homogenize* **h**e*te*ro**ge**neo***us*** datasets for training a federated learning model and determining necessary **granularity** for accurate model performance.

Machine Learning vs Federated Learning



Traditional Machine Learning

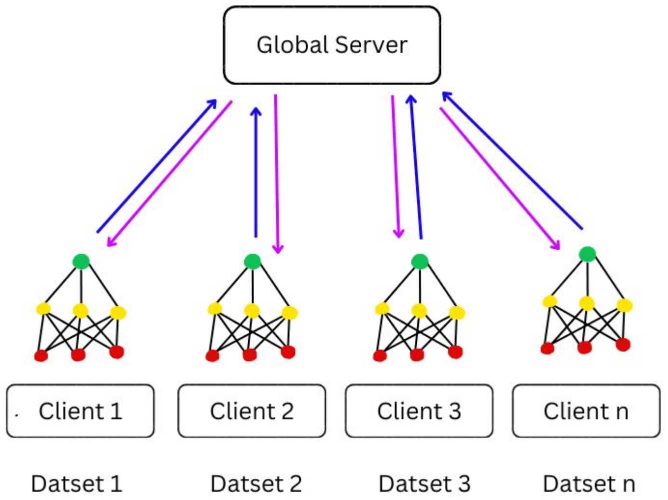


Federated Learning Architecture

UMBC

Federated Learning (FedML)

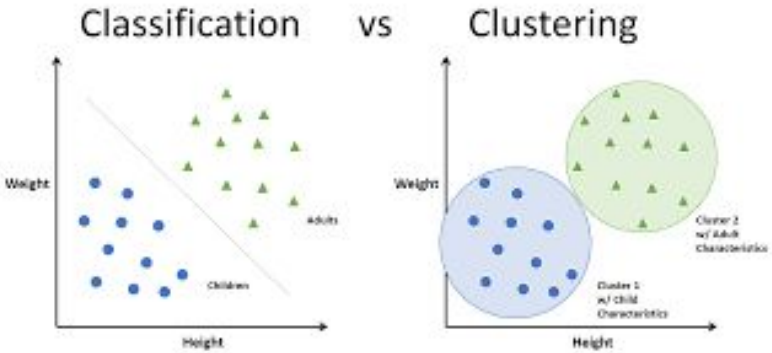
- Advantage
  - Protects user privacy
    - Sends model weights
- Disadvantage
  - Slower weight updates
    - Slower convergence
      - Minimize the loss function
  - Sensitive to heterogeneity
    - Datasets



UMBC

Statistical Heterogeneity

- Causes
  - Skewed label distribution
  - Skewed feature distribution
  - Granularity differences



- Approach
  - Ignore annotations and recluster based on images
  - Use annotations to confirm reclustering
  - Must determine number of classes

Methodology

- Annotated Ship Datasets
  - ABOShips, Seaships, VIS onshore and offshore
- Python Scripts:
  - Crop and Sort Images
  - Extract Features
  - Create T-SNE Plot
    - Determine Perplexity Value
  - Recluster Images & Reannotate
- Future: Use to train model

```python
# vector extractor
from ast import main
import sys
import os
sys.path.append("../img2vec_pytorch")  # Adds higher directory to python mod
from img_to_vec import Img2Vec
from PIL import Image
import random
import numpy as np
from sklearn.manifold import TSNE
from matplotlib import cm
import matplotlib.pyplot as plt

# Takes random images from the folder it is given
def get_random_images(folder_path, num_images):
    images = os.listdir(folder_path)
    random.shuffle(images)
    return images[:num_images]

def scale_to_01_range(x):
    # compute the distribution range
    value_range = (np.max(x) - np.min(x))

    # move the distribution so that it starts from zero
    # by extracting the minimal value from all its values
    starts_from_zero = x - np.min(x)

    # make the distribution fit [0; 1] by dividing by its range
    return starts_from_zero / value_range


#should loop through the folders of the path given and take two random image
#calculate their median vectors and store them in a dictionary
def make_vector_dictionary(main_folder_path, num_images):
    img2vec = Img2Vec()
    median = 0
    vector_cumulative={}
    folders = os.listdir(main_folder_path)
    for folder in folders:
        vectors = []
        images = get_random_images(os.path.join(main_folder_path, folder), n
        for image in images:
            image_path = os.path.join(main_folder_path, folder, image)
            print("image path: ",image_path)
            img = Image.open(image_path).convert('RGB')
            vec = img2vec.get_vec(img) #[[]] - one image
            vectors.append(vec) #[[[]],[[]],[[]],[[]]] - one class
```

UMBC

Dataset Preparation: Cropping

- ABOShips, Seaships, VIS offshore and onshore
- Annotations
  - Seaship boundaries
    - X min
    - X max
    - Y min
    - Y max
  - Boat Class
    - Ex: cargo ship, passenger ship, cruise-boat, bulk cargo carrier
- Crop and categorize into class folders


```
{
    "boxes": [
        {
            "label": "ore carrier",
            "x": 1147.5,
            "y": 448,
            "width": 733,
            "height": 104
        }
    ],
    "height": 1080,
    "key": "000101.jpg",
    "width": 1920
}
```

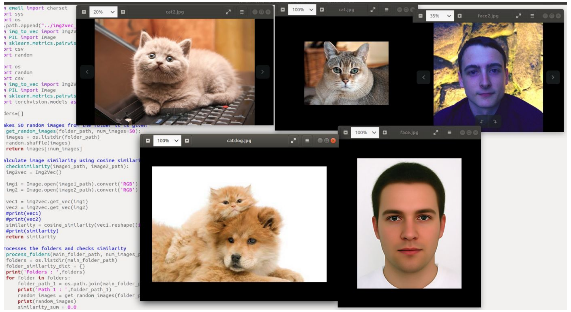

UMBC

Dataset Preparation: Feature Extraction

- Convert images to vectors based on averaging feature vectors the algorithm recognizes and extracts
  - Stores the features as a numpy array
- Off the shelf resnet feature extractor (CNN)
  - Github repository: img2vec
  - Fixed classes
- Allows direct numerical image comparison
  - Similarity score csv files
- Customized Python Scripts

	Standard	Standard	Standard
1	Folder 1	Folder 2	Similarity
2	Boat	Miscellaneous	0.97630334
3	Boat	Passengership	0.8921411
4	Boat	Motorboat	0.97385085
5	Boat	Ferry	0.8302501
6	Boat	Militaryship	0.7427224
7	Boat	Miscboat	0.9201295
8	Boat	Cruiseship	0.77406377
9	Boat	Sailboat	0.6982449
10	Boat	Seamark	0.9686703
11	Boat	Cargoship	0.764315
12	Miscellaneous	Boat	0.97630334
13	Miscellaneous	Passengership	0.8277408
14	Miscellaneous	Motorboat	0.9299425

aboships_similarity.csv

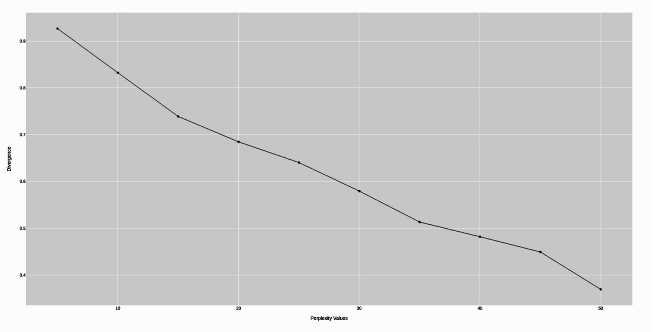


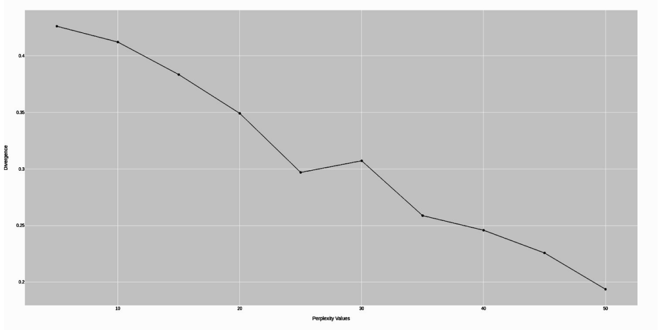
img2vec simulation

UMBC

T-SNE Plot

- t-distributed stochastic neighbor embedding (T-SNE)
- Nonlinear dimensionality reduction algorithm to reduce dimensionality
  - Clusters similar points together and distance between different clusters
- Perplexity value
  - If low, tendency is too many points together in a cluster & will not increase distance between different clusters
  - If high, opposite occurs
- Perplexity vs Divergence Graphs: pinpoint correct value
  - Divergence quantifies the difference between 2 probability distributions (ie clusters)
  - We find the minimum divergence before stabilization and take its perplexity value



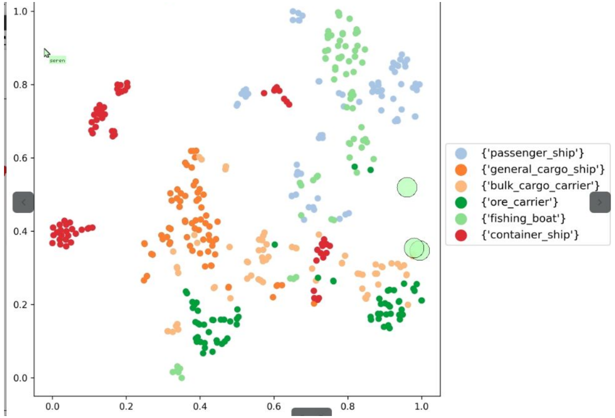


Perplexity vs Divergence Graphs Seaships & ABO Ships

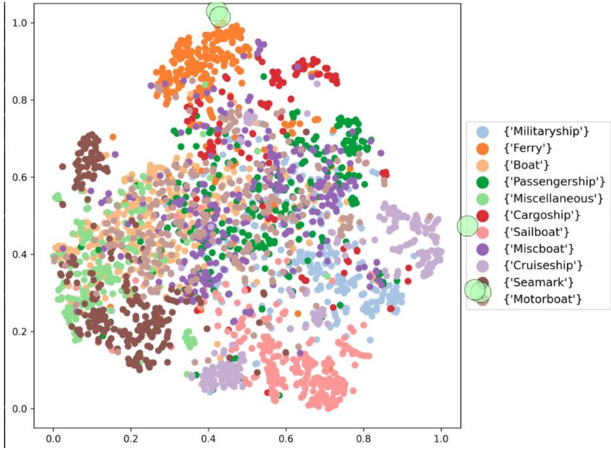
UMBC

T-SNE Dataset Visualization

- We group ships based on features
- T-SNE allows cluster visualization of similarities/differences between classes
- Python Script
- Set perplexity level to previously determined values



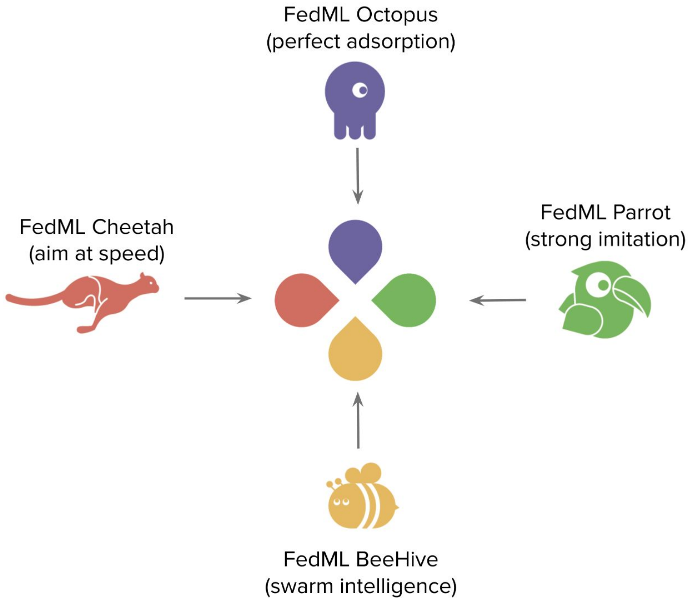
tsne_seaships_p25_im70



tsne_ABOships_p17_im300

Future Steps

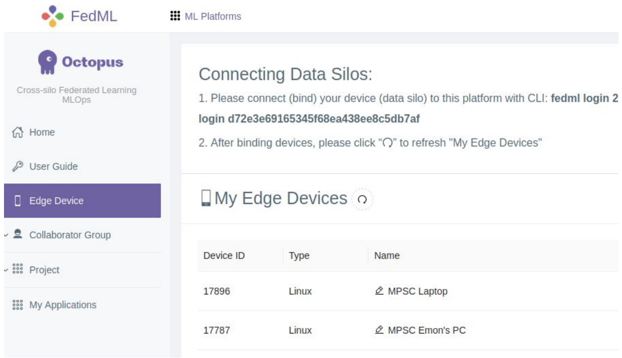
- Cluster the datasets together
- Apply method to the federated learning setting
  - Integrate with FedML platform cross-silo edge devices
- Impact: can be applied to preparing many different types of image datasets
  - Is usable strategy for homogenization

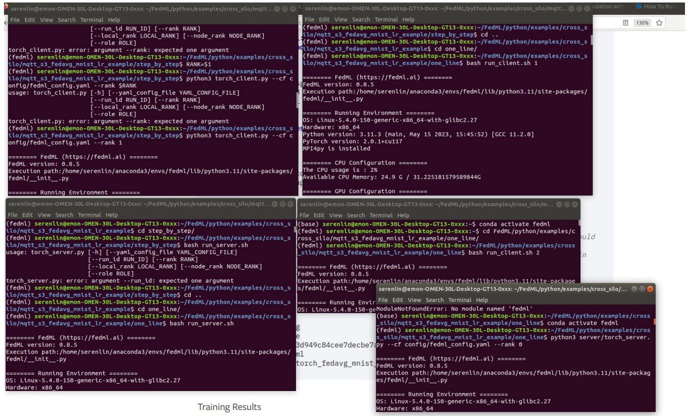


UMBC

Skills Learned Specific to Project

- Fundamentals of Machine & Federated Learning
  - Math behind the models: gradient descent algorithms, convolutional neural networks (cnn), loss functions. Back batch propagation, feature selection, unsupervised/supervised learning, bias-variance tradeoff
- Ubuntu Linux Terminal
  - Install and execute programs and code
- FedML Simulations and ML-ops Platform
- Github
- Python
  - Libraries: tensorflow, pytorch, sklearn, matplotlib
  - File image cropping, feature extraction, t-sne plot creation, perplexity scores, csv file read and write



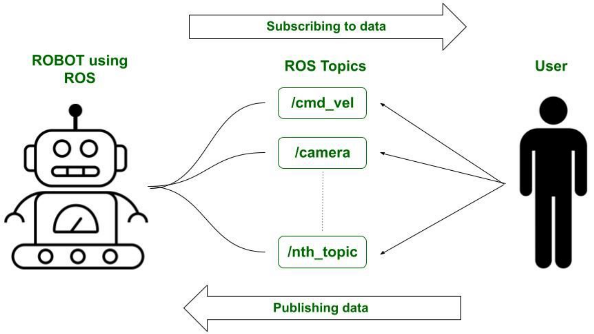


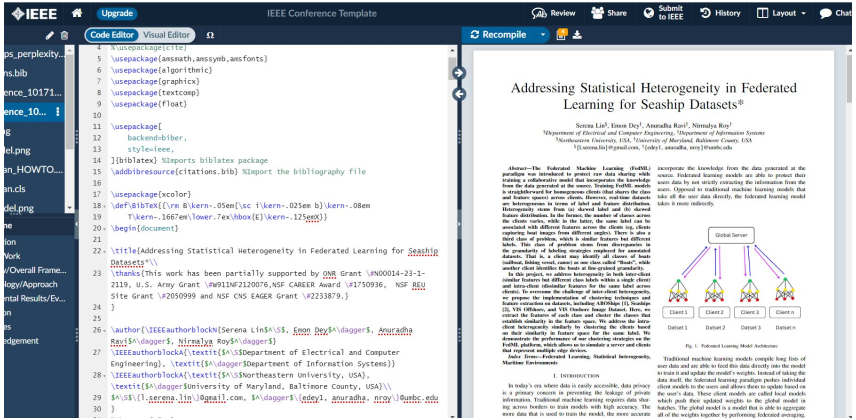
Training Results

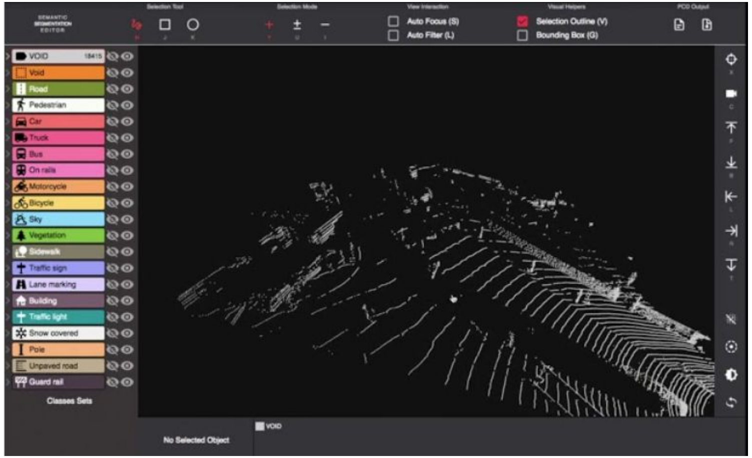
UMBC

Other Research Skills Learned

- Robot Operating System (ROS)
  - Fundamentals, writing publisher and subscribers in c++ and python
- Google Colab: keras machine learning model
- Semantic Segmentation Editor: Lidar
- Overleaf: LaTeX
  - Documentation & IEEE Paper Formatting







UMBC

References

[1] B. Iancu, V. Soloviev, L. Zelioli, and J. Lilius, Aboships – an inshore and offshore maritime vessel detection dataset with precise annotations, 2021. arXiv: 2102 . 05869

[2] Z. Shao, W. Wu, Z. Wang, W. Du, and C. Li, “Sea-ships: A large-scale precisely annotated dataset for ship detection,” IEEE Transactions on Multimedia, vol. 20, no. 10, pp. 2593–2604, 2018.: 10.1109/TMM.2018.2865686.

[3] J. Wang, Z. Charles, Z. Xu,et al., A field guide to federated optimization, 2021. arXiv: 2107.06917

[4] A. Ahmad, W. Luo, and A. Robles-Kelly, “Robust federated learning under statistical heterogeneity via hessian spectral decomposition,” Pattern Recognition, vol. 141,p. 109 635, 2023,ISSN: 0031-3203.DOI: https : / / doi org/10.1016/j.patcog.2023.109635. [Online]. Available:https : / / www . sciencedirect . com / science / article / pii /S0031320323003369.

[5] J. Han, A. F. Khan, S. Zawad,et al., “Heterogeneity-aware adaptive federated learning scheduling,” in 2022 IEEE International Conference on Big Data (Big Data),2022, pp. 911–920 10.1109/BigData55660.2022.10020721.

[6] B. Gong, T. Xing, Z. Liu, J. Wang, and X. Liu, “Adaptive clustered federated learning for heterogeneous data in edge computing,” Mob. Netw. Appl., vol. 27, no. 4, pp. 1520–1530, Aug. 2022,: 1383-469X.: 10.1007/s11036- 022- 01978- 8. [Online]. Available: https: //doi.org/10.1007/s11036-022-01978-8.

[7] M. Er, Y. Zhang, J. Chen, and W. Gao, “Ship detection with deep learning: A survey,” Artificial Intelligence Review , pp. 1–41, Mar. 2023.: 10 . 1007 / s10462 - 023-10455-x.

[8] C. He, S. Li, J. So, et al., “Fedml: A research library and benchmark for federated machine learning,”ArXiv vol. abs/2007.13518, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:220793772.

Acknowledgements

- Emon Dey
- Dr. Anuradha Ravi
- Dr. Nirmalya Roy
- PhD Students in the Lab
- Ms. Marjory Pineda
- Fellow REU students

This work has been partially supported by ONR Grant #N00014-23-1-2119, U.S. Army Grant #W911NF2120076,NSF CAREER Award #1750936, NSF REU Site Grant #2050999 and NSF CNS EAGER Grant #2233879.

UMBC

Q & A

UMBC