

Nirbhay Sharma | +91 9369630713 | sharma.59@iitj.ac.in | [Github](#) | [Portfolio](#) | [LinkedIn](#)

Education

B.Tech, CSE | Indian Institute of Technology (IIT) Jodhpur | CGPA: 8.97/10

08/2019-05/2023

Technical Skills

Programming Languages: Python, C/C++ | **Skills:** Machine Learning, Deep Learning, Computer Vision

Tools and Frameworks: Pytorch, Django, Flask, Docker, AWS Lambda, Regex, Git, Github, Firebase, MySQL

Publications

Nirbhay Sharma, Gautam Kumar, Dr. Angshuman Paul, **An Extremely Lightweight CNN Model For the Diagnosis of Chest Radiographs in Resource-constrained Environments** | *International Journal of Medical Physics* 2023 | [Paper](#)

Research and Work Experience

Faaya Astu India | Full Time (ML Engineer) | Pytorch, Diffusion, ControlNet, LoRA

06/2023-Present

- Worked on **SOTA Stable Diffusion, ControlNet** and **LoRA** models for text to print, and content generation
- Trained **Stable Diffusion ControlNet** architecture on **Lineart** and **Colorbox** control on **VastAI** GPU instance to provide more flexibility and control on print generation
- Trained **Low Rank Adaptation (LoRA)** models using **Kohya_SS** for controlled background and face generation
- Containerized **Stable Diffusion WebUI** and **ControlNet** using **Docker** to deploy them as **Serverless Endpoints** on **RunPod** and exposed endpoint APIs to **AWS Lambda** to create **APIs for APP** using **AWS API gateway**

ExaWizards India | Intern (ML Engineer) | Split Neural Networks | Pytorch, Jetson Nano, PySyft

06/2022-07/2022

- Splitted **Mask-RCNN, FCN_Resnet50, YOLOv5** for **Instance segmentation, segmentation, face detection** tasks
- Implemented **Autoencoder** model for efficient **image compression to latent space** and setup **Pysyft** to communicate latent features from **Jetson Nano** to GPU server, **preserving data privacy** at Jetson Nano

Noise Engineered Federated Distillation for Heterogeneous Settings | Pytorch, Federated Learning, Data-Free KD

Research Project | Supervisor: Dr. Deepak Mishra | IIT Jodhpur

08/2022-05/2023

- Proposed a **novel Federated Learning (FL) Framework** to handle **model and data heterogeneity**
- Implemented **data-free KD** using **Gaussian Noise** at the **Server**, eliminating need of **proxy dataset** or **GAN's**
- Our algorithm achieved **3x better performance** compared to SOTA FL algorithms like **FedDF, FedMD, Kt-pfl**

Light Weight CNN Model for Chest Radiographs Classification | Pytorch, Light Weight CNN Models

Research Project | Supervisor: Dr. Angshuman Paul | IIT Jodhpur | [Paper](#)

06/2021-03/2022

- Designed a **Lightweight CNN model (ExLNet)** for the abnormal detection of **Chest Radiographs**
- Fused **Squeeze and Excitation** blocks with **Depth-wise convolution** to create **DCISE** layer as a component of **ExLNet**, which outperforms SOTA models like **Mobilenet, Shufflenet** on medical datasets like **NIH, VinBig**

Projects

Regularizing Federated Learning (FL) via Adversarial Model Perturbations (AMP) | [Github](#) | [Report](#) | Pytorch, FL

- Analyzed the effect of **Adversarial Model Perturbations (AMP)** on **4 SOTA Federated Learning (FL) algorithms**
- Implemented **FedAvg, FedProx, FedNTD, SCAFFOLD** from **scratch** and integrated **AMP** module at the **client**
- Observed a boost of **2-3% accuracy** on **CIFAR10/100** dataset after integrating **AMP** in each of the algorithm

Image Captioning using Detection Transformer | [Github](#) | Pytorch, DeTR, Transformer

- Implemented **DeTR** from **scratch** using **Pytorch** and modified it for **image captioning** task
- Trained **DeTR** on **Flickr30k** dataset for **500 epochs** and achieved a **BLEU score of 57.36** on **Flickr8k** dataset

Transformers-Implementation | [Github](#) | Pytorch, Transformer

- Read and Implemented **11 research papers** on **vision transformers** from scratch in Pytorch
- Implemented SOTA models like **Swin Transformer, Pyramid ViT, Convolution ViT** etc. for various vision tasks such as **Image classification, Object Detection, Sementic Segmentation**