Dissecting the Origins of the Power Side-Channel in Deep Neural Networks

Sayed Erfan Arefin;
Abdul Serwadda.

Texas Tech University

Department of Computer Science

IEEE AIIoT Conference 2024

Introduction to DNNs and Power Analysis

- Deep Neural Networks (DNNs) powers advanced applications: Fraud detection, Computer vision etc.
- DNNs are substantial commercial assets requiring significant investment.
- The architectures of DNNs represent critical intellectual property.
- DNNs creates power consumption signatures.
- Power consumption analysis can be used to infer architectural details of DNNs.

Related works

- Hardware based measurements:
  - Méndez et al. conducted side channel attack on embedded DNNs in IoT.
  - Nagarajan et al. examined DNNs' susceptibility to side-channel attacks.
  - Batina et al. investigated neural network vulnerabilities on ARM based micro-controllers.
- Software based measurements: Our previous work focuses on inferring DNN architecture via power side-channel, using GPU-z software.

Our work: Study Tools

- GPU sensors
  - GPUs are equipped with current and voltage sensors.
  - Used to maintain balance between power-consumption and temperature.
- We utilized **Nvidia-smi** tool
  - A native tool included with Nvidia drivers.
  - High resolution appx. 1000 samples/second
- Benefits:
  - Eliminates the need for physical access.
  - Eliminates the need for additional software.

Experiment Design: Types of Experiments

- Software base power side channel attack:
  - We assume attacker gained access to system.
  - Attacker collects power data with nvidia-smi.
  - We focus on Convolutional Neural Networks.
  - Infer CNN model while running image classification on the GPU.
- Analysis of CNN modules or units:
  - CNNs are composed of convolutional, pooling, and fully connected layers, dropout and batch normalization
  - Analyze power signature for these modules.

Experiment Design: Data Collection

- Captures power consumption data and stored in CSV.
  - 100 samples/sec for architecture inference
  - 1000 samples/sec for CNN units analysis
- Power measurement is conducted when the CNN image classification tasks are running.

Experiment Design: Test Dasets

- For architecture inference attack, 80 samples for training, 20 for testing across two distinct datasets from imagenet.
  - Dataset 1: Cats and Dogs Dataset (50 images each).
  - Dataset 2: Random images (100 images).
- CNN units analysis was conducted using random tensors across typical input sizes.
  - Non-Fully Connected Layers: Common sizes include 32x32, 64x64, 128x128, 256x256, and 320x320 pixels.
  - Fully Connected Layers: Typically utilize input sizes of 196, 392, 784, 1568, and 3136.

Experiment Design: Target Machines

GPU model	General Information		Architecture	Power	Number of cores		Memory Config	Collected from
	Usage	Year			Tensor	Cuda		
GTX 1070 Ti	Home use	2017	Pascal	180 W	0	2432	8 GB GDDR5	Physical Access
RTX 2060 super	Home use	2019	Turing	175 W	272	2176	8 GB GDDR6	Physical Access
Quadro RTX 4000	Data-center	2018	Turing	160 W	288	2304	8 GB GDDR6	Remote access
Quadro RTX 5000	Data-center	2018	Turing	265 W	384	3,072	16 GB GDDR6	Remote access
Tesla v100	Data-center	2017	Volta	250 W	640	5120	32GB HBM2	Remote access
RTX A40	Data-center	2020	Ampere	300 W	336	10752	48GB GDDR6	Remote access

Table: GPU specification, usage, and access type in our experiments

CNNs and their variants

- **SqueezeNet:** SqueezeNet 1.0, SqueezeNet 1.1
- **DPN:** DPN68, DPN68B etc.
- **DenseNet:** DenseNet121, DenseNet161 etc.
- **VGG:** VGG11, VGG13, VGG16 etc.
- **ResNet:** ResNet18, ResNet50, CaffeResNet101 etc.
- No variants: AlexNet, XceptionNet, PolyNet etc.

Experiment Design: Experiment Configurations

1. **Configuration-1: Core Neural Networks**
   - Focus on broader family classification for models with variants.
   - Attacker has knowledge of some variants of the family.
   - Identification of 11 core CNN models.
2. **Configuration-2: Variant Classification**
   - Specific variant identification within a known neural network family.
   - Challenges include 4-class, 5-class, and 10-class for variants in DenseNet, DPN, and ResNet.
3. **Configuration-3: Comprehensive Classification**
   - Classification without prior knowledge of the network family.
   - A 35-class problem including all core models and their variants.

Studying the Power signatures of the units

- Convolution Layer power consumption for different inputs.

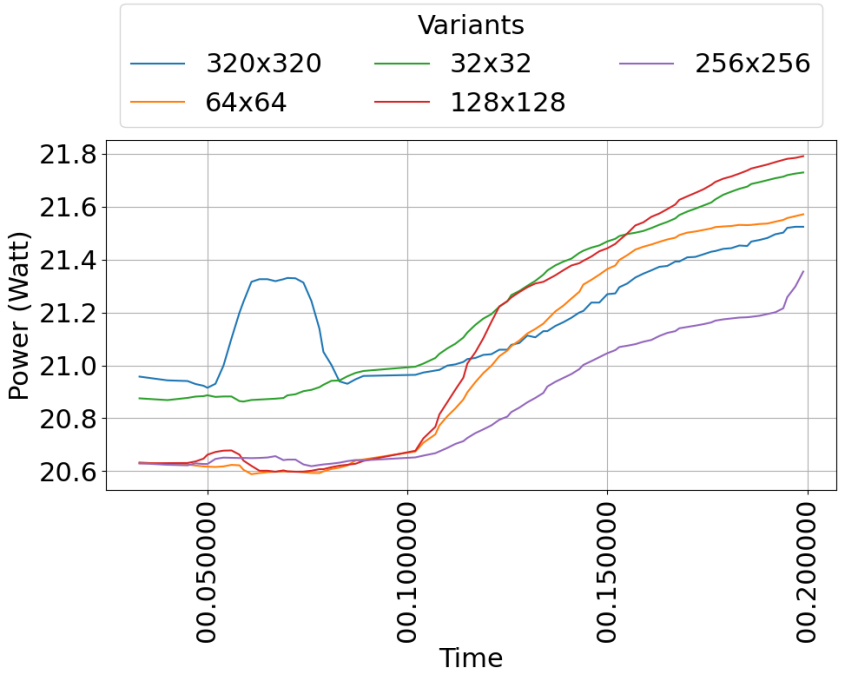
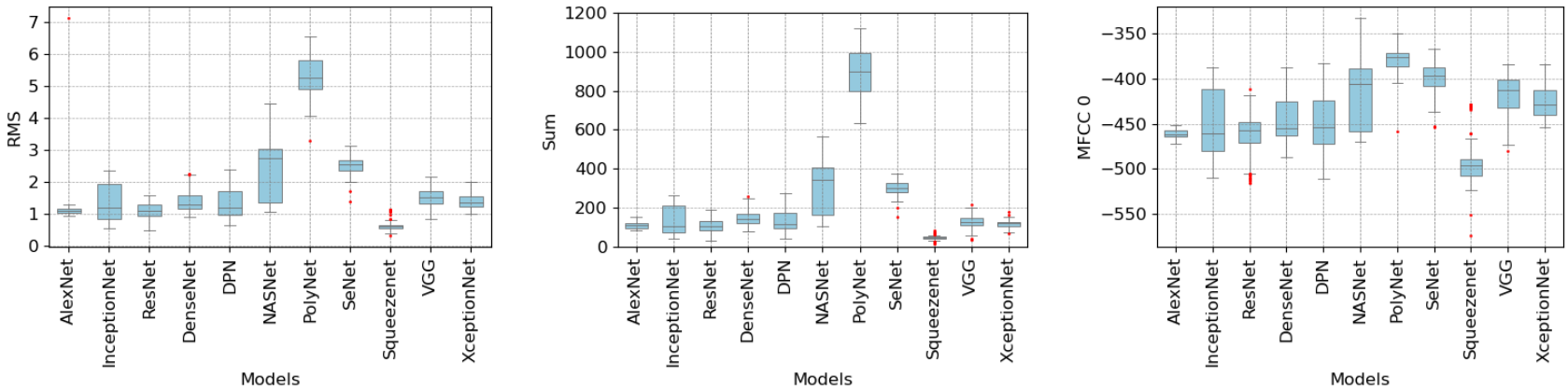


Figure: Convolution Layer

Feature Exploration



(a) Root mean square (b) Sum (c) MFCC 0

Figure: Some features values for RTX 2060 from the results of Scikit learn's k-best (10 best) features, used in Experiment config-1 collected on dataset-2.

Results: Classification

- **Featuer Extraction:** Extraction of statistical and spectral features using Numpy and Librosa libraries.
- **Classification Algorithms:** Employed Random Forest, XGBoost, and Light GBM.
- **Model Tuning:** Used Sklearn pipeline from Scikit Learn for fine-tuning models.
- **Feature Selection:** Features selected using Sklearn's Select K-Best method to choose the top 25 features.
- **Optimal Configurations:** Normalizer and Max Absolute Scaler emerged as effective choices for maximizing test accuracy.

Results: Configuration 1

- Experiment Config-1 classification accuracy.
- Best results were produced on Nvidia 1070 Ti
  - Using XGBoost
  - Dataset 1: 94.00%
  - Dataset 2: 95.29%

Device	Classifier	Dataset 1	Dataset 2
Nvidia 1070 Ti	Light GBM	94.67%	94.41%
	Random Forest	94.33%	95.29%
	XGBoost	94.00%	95.29%
Nvidia RTX 2060 Super	Light GBM	71.76%	73.24%
	Random Forest	71.76%	71.76%
	XGBoost	70.29%	73.24%
Nvidia A40	Light GBM	64.12%	70.00%
	Random Forest	62.94%	67.94%
	XGBoost	67.06%	69.71%
Nvidia RTX Quadro 4000	Light GBM	68.82%	69.71%
	Random Forest	63.53%	67.94%
	XGBoost	68.24%	69.71%
Nvidia RTX Quadro 5000	Light GBM	68.53%	78.24%
	Random Forest	64.71%	75.00%
	XGBoost	68.53%	76.76%
Nvidia Tesla V100	Light GBM	86.76%	86.47%
	Random Forest	86.47%	85.29%
	XGBoost	87.06%	84.12%

Results: Configuration 2 & 3

- Config-2: **ResNet variants** classification (10-class problem)
- Best results were produced on Nvidia 1070 Ti
  - Using Random forest
  - Dataset 1: 79.00%
  - Dataset 2: 80.00%
- Config-3: All variants classification (35-class problem)
- Best results were produced on Nvidia 1070 Ti
  - Using Random forest
  - Dataset 1: 85.67%
  - Dataset 2: 87.33%

Mitigation: Inject background noise

- **Mitigation through Noise:** Introduce noise into power data during CNN operations.
- **Random Computation:** Random matrix multiplications in GPU.
- **Experiment Setup:** Matrix size randomly varies from 2 to n, running experiment configuration 1 on Dataset 2.
  - **Noise Levels:** Different intensities (powers of 2) tested on Nvidia RTX 2060 Super GPU.
  - **Impact on Accuracy:** Increase noise until classifier accuracy approximates random guessing.

Mitigation: Inject background noise

- Background noise injection
- Experiment config-1
- Conducted on,
  - Dataset-2
  - RTX 2060

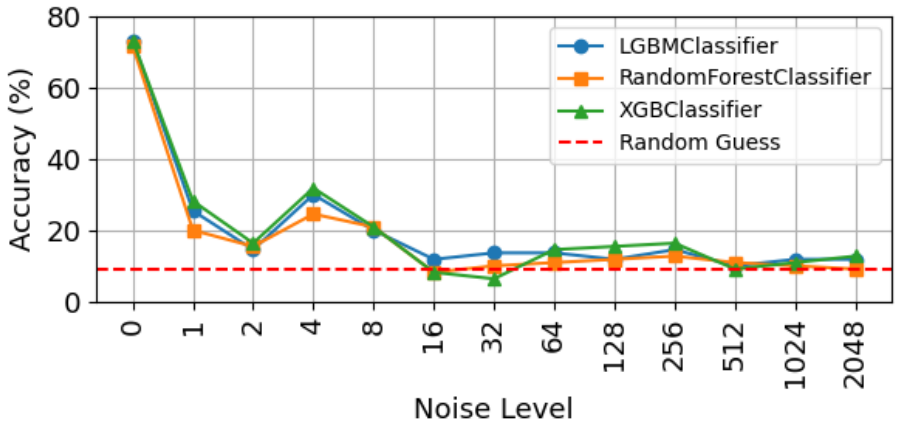


Figure: Classifier accuracy for different noise levels.

Impact on power consumption

- Avg. power consumption for different noise levels
- Experiment config-1
- Conducted on,
  - Dataset-2
  - RTX 2060

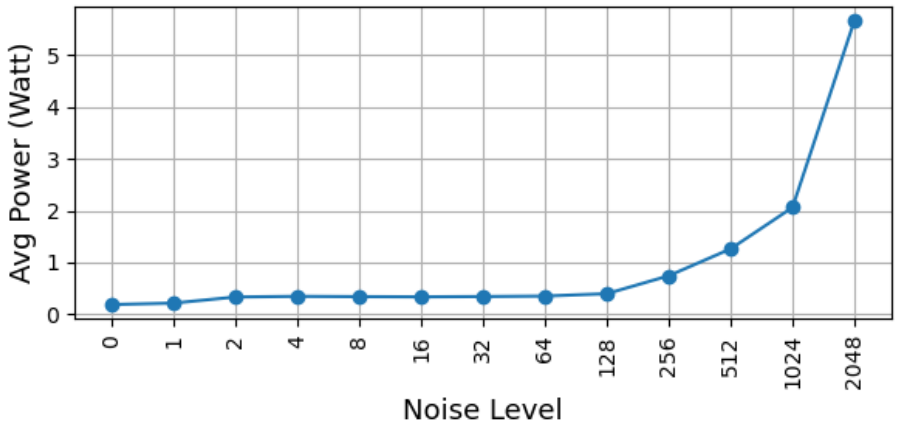


Figure: Avg. power consumption for different noise levels.

Mitigation: Disable power measurement

- **Restrict Hypervisor-Level Commands:**
  - Implement restrictions on executing Nvidia-smi commands at the hypervisor level for running nvidia-smi.
- Effectiveness in Cloud Settings:
  - This approach is particularly effective in cloud environments where hypervisors control resource access.
- Limitations on Desktop Systems:
  - Less viable on desktop systems, where users typically have easy access to the resources.

Conclusion

- Conducted a detailed analysis of software-based power side channels using Nvidia-smi.
- Focused on power consumption patterns across core neural network components.
- Explored identification of different CNN architectures based on power data.
- Performed experiments on various devices, including physical and cloud-based GPUs.
- Developed strategies for mitigation of such attacks.

Thank You!

Have any questions?