UF Herbert Wertheim College of Engineering
UNIVERSITY of FLORIDA

Nuclear Engineering Program
Materials Science and Engineering Department

Prediction of PWR Pin Powers using Convolutional Neutral Networks

Scientific Machine Learning for Nuclear Engineering Applications Workshop
PHYSOR 2022, May 15 – 20

Justin K. Watson, Forrest Shriver, Aidan Furlong

Acknowledgements

- University of Florida internal funding
- NEUP Fellowship
- University of Florida HiPerGator
- This research used resources of the Compute and Data Environment for Science (CADES) at the Oak Ridge National Laboratory, which is supported by the Office of Science of the US Department of Energy under Contract No. DE-AC05-00OR22725.
- This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

Dr. Forrest Shriver

Sentinel Devices

Forrest Shriver is currently the CEO of Sentinel Devices, an early-stage startup focused on improving equipment maintenance and reliability via distributed in-the-field artificial intelligence. His company is currently funded by a $500,000 fellowship and grant from the U.S. Department of Energy.

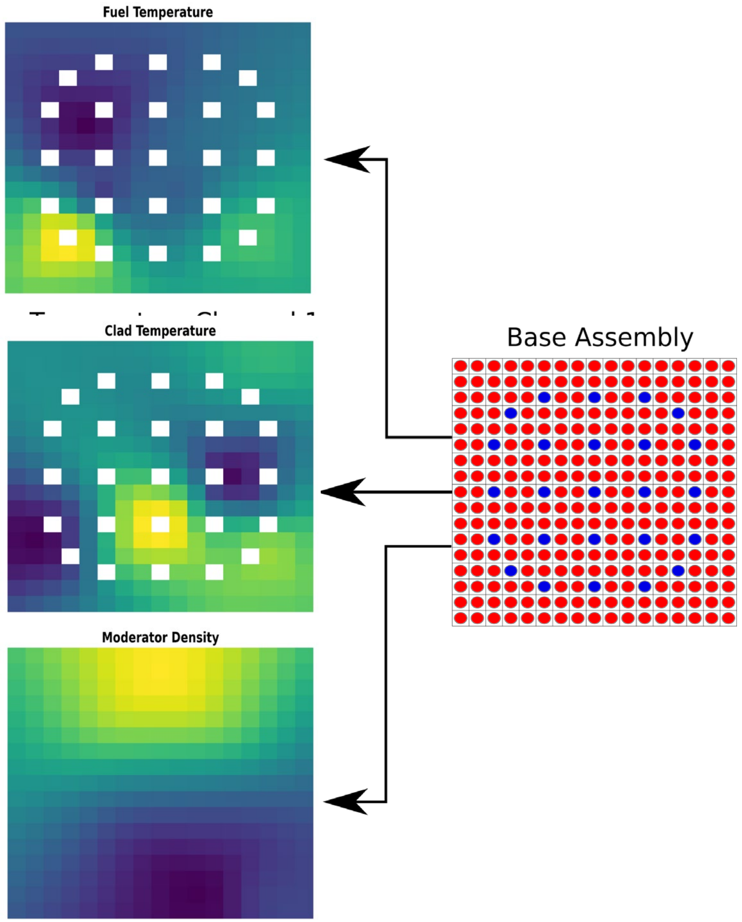
Aidan Furlong

Aidan is an undergraduate student currently in his senior year. He plans on getting a Master’s degree after graduation.

Motivation, Goals, and Overview

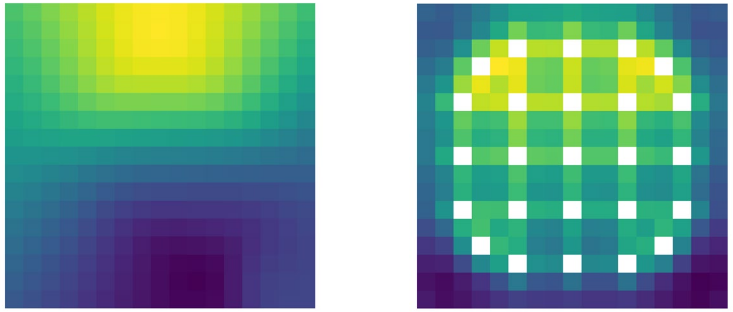
Goals

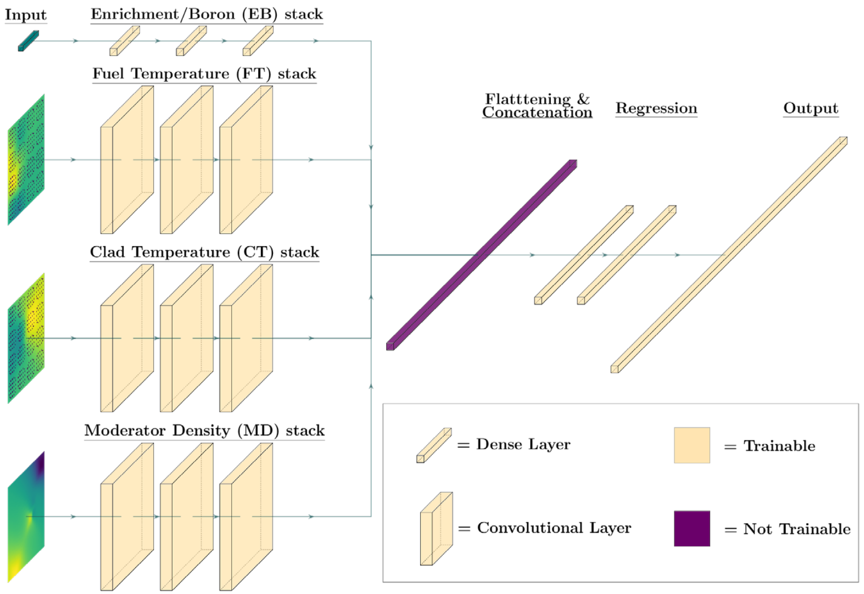
1. Develop machine learning surrogate models to supplement expensive high-fidelity core simulators.
   - High-fidelity solution method acceleration (single physics and multi-physics)
   - Augment nodal methods (pin power reconstruction)
   - Fuel Cycle Optimization
   - Feedback modeling for fuel performance applications
2. Can produce model with accuracy of lattice code and runtime of nodal code.
3. Produce approximate models rapidly w/o spending human effort fine-tuning.
4. Produce models that are robust against out-of-bounds data.



Overview

- Developed a methodology for the application of deep learning to light water reactor (LWR) modeling & simulation.
- Developed a general architecture (LatticeNet) for predicting high-fidelity parameter distributions (pin powers) in both single and multi-assembly regions.
  - Based on computer vision methods
- Generate a database of high-fidelity inputs/outputs
  - Pin (or sub-pin) inputs correlated to target parameters
- Developed a method to determine if a target model will fail to give a physically realistic answer – even for regimes with no training data
- Explored methodology generality to novel distributions separate from the training distribution

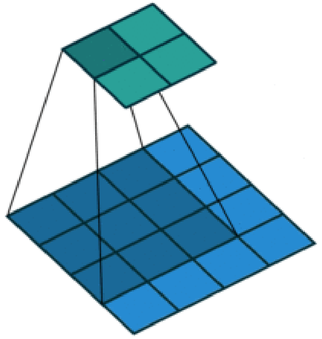




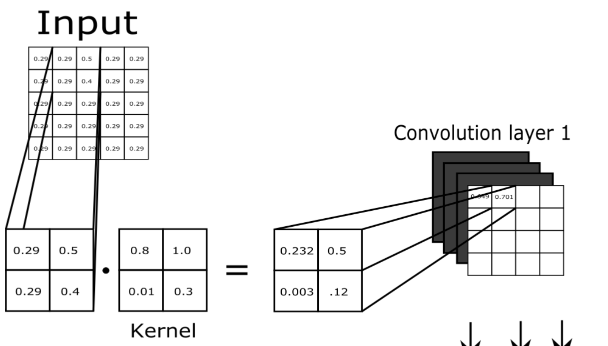
LatticeNet

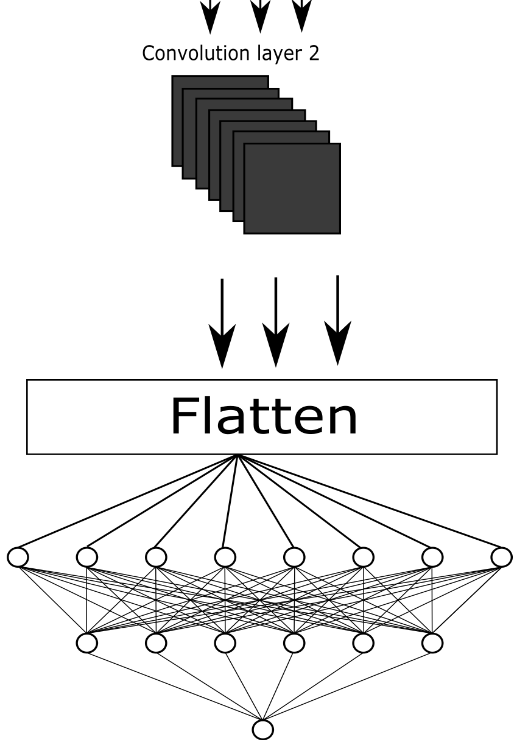
Methods – Neural Networks: Convolutional NN’s

- Proposed initially to perform digit recognition
- The most well-known form (2D convolution) imposes the constraint that data must be in a two-dimensional input format
- Applies a kernel across every spatial portion of the input image
- In order to capture different features, multiple $M \times N$ filters are applied in a single “layer” of a CNN; stacks of successive convolutional layers are often called convolutional stacks
- Extremely common in image processing/computer vision tasks as they allow the easy learning of data-based priors, instead of having to hand-craft features yourself



V. Dumoulin, F. Visin, “A guide to convolution arithmetic for deep learning”

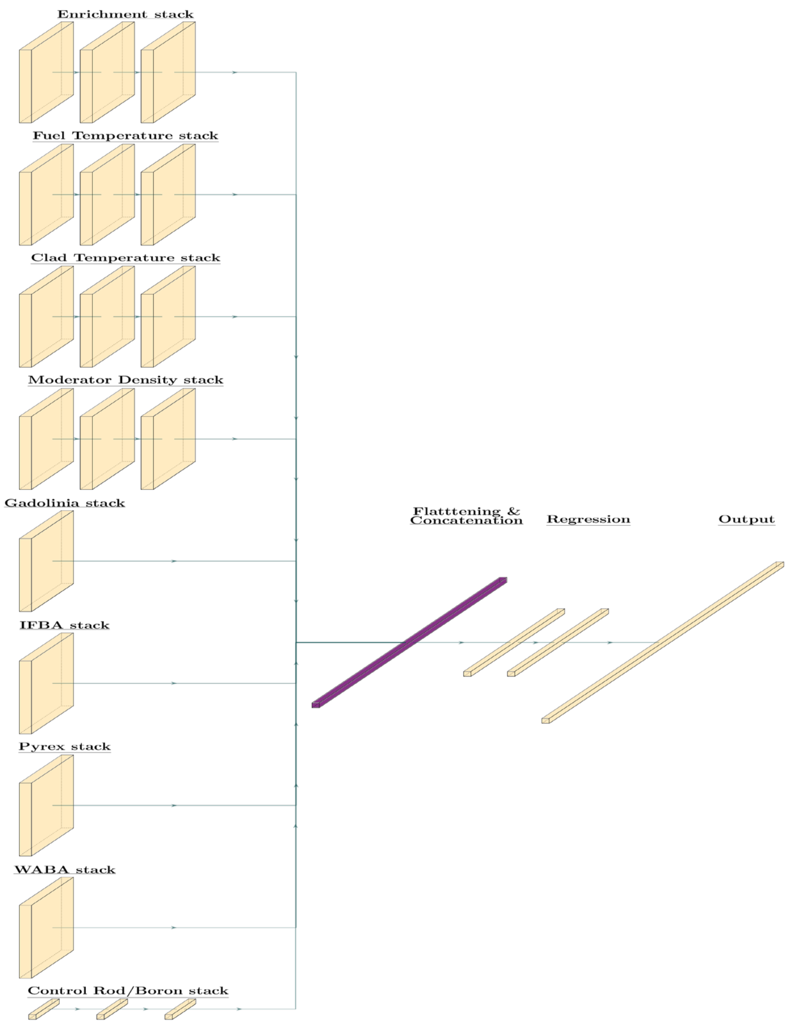




LatticeNet – Datasets

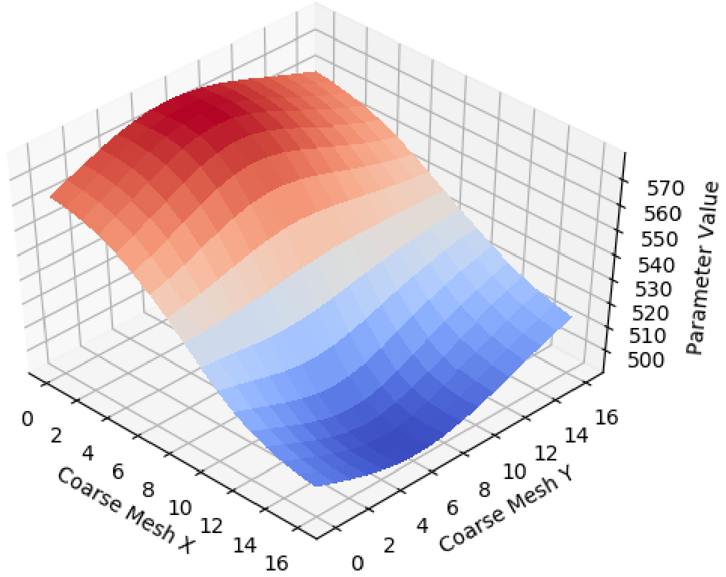
- Started with a single, reflective 2D PWR assembly
  - U-235 enrichment is allowed to vary between 1.8 and 4.9%
  - CR position fully in or fully out
  - Boron allowed to vary between 0-2,000 ppm
  - All TH inputs (moderator density, fuel temperature, clad temperature) randomly varied independently according to the methodology described
  - 4,050 random assemblies generated using this methodology
- Additional sets of 4,050 were generated including burnable poisons (BPs)
  - Gadolinia
  - IFBA
  - Pyrex
  - WABA

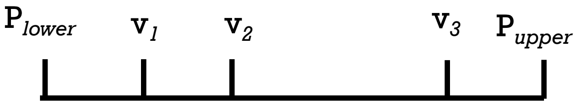
Design parameter	Allowed Range
Fuel temperature (Celsius)	286–1,326
Cladding temperature (Celsius)	286–356
Moderator density (g/cc)	0.660–0.743
Fuel Enrichment	1.8%–4.9%
Control Rod position	0, 1
Boron Concentration (ppm)	0 – 2,000



Methods – TH Variation

- Don't necessarily want to drive our simulations using realistic TH inputs from iterative solves
  - Too expensive, have to discard half our CPU cycles
  - Not guaranteed to be useful as physical training data (mostly the same)
- Solution: Randomly sampling continuous TH input curves
  - Pick vertices with random location and magnitude between P_{upper} and P_{lower}
  - Every other value is a weighted sum, weight a power relationship of inverse distance

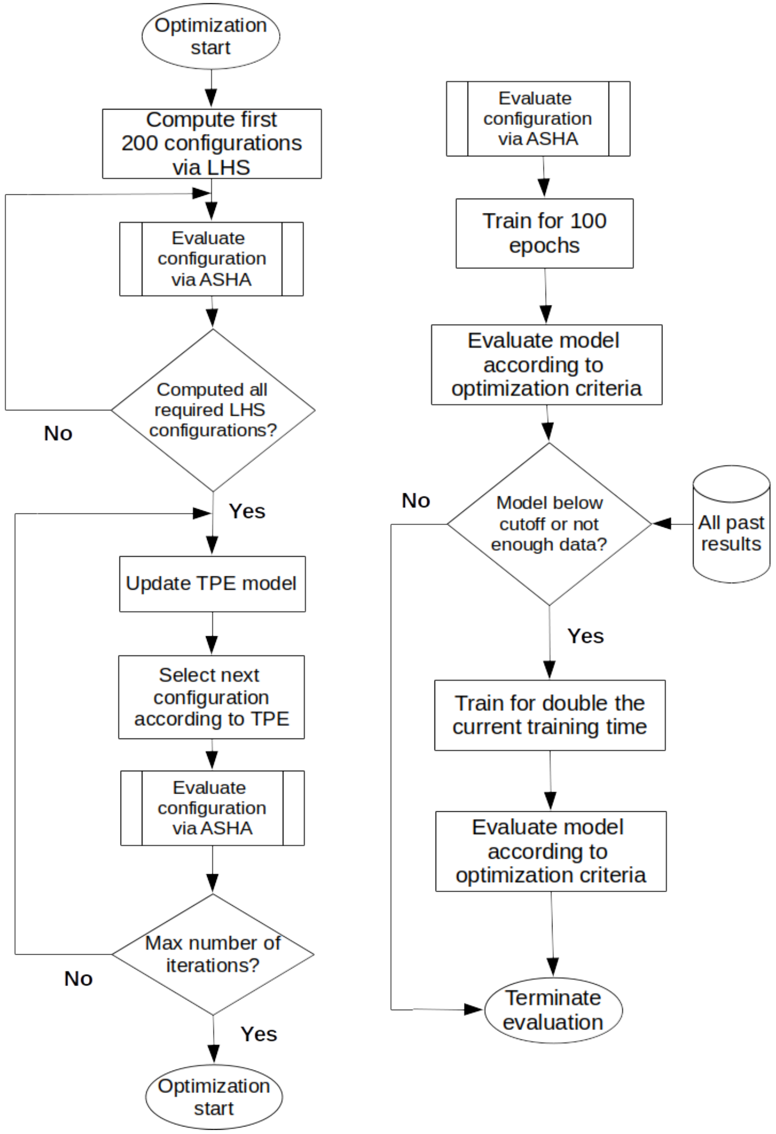




$$P = \frac{\sum_i^N w_i v_i}{\sum_i^N w_i} \qquad w_i = \frac{1.0}{d_i^m}$$

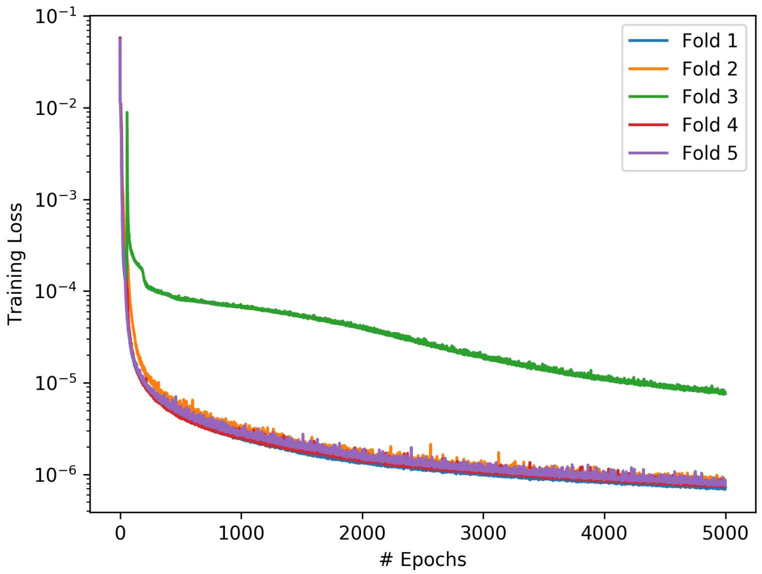
Methods – Hyperparameter Optimization

- Network performance depends on hyperparameter choice
  - Problems can be very sensitive to particular choices at particular stages
  - No existing method to analytically determine best hyperparameter choices
  - Large body of existing research and past choices for mainstream applications, however little/no past choices for our application
  - Not intuitive we can make the same choices converged to in other fields
- Solution: Develop a method of hyperparameter optimization that is inexpensive and allows us to converge to good hyperparameters quickly
  - Latin Hypercube Sampling (LHS) to initially understand the search space
  - Tree Parzen Estimators (TPE) to suggest good subsequent hyperparameters
  - Adaptive Successive Halving Algorithm (ASHA) to early stop bad trials



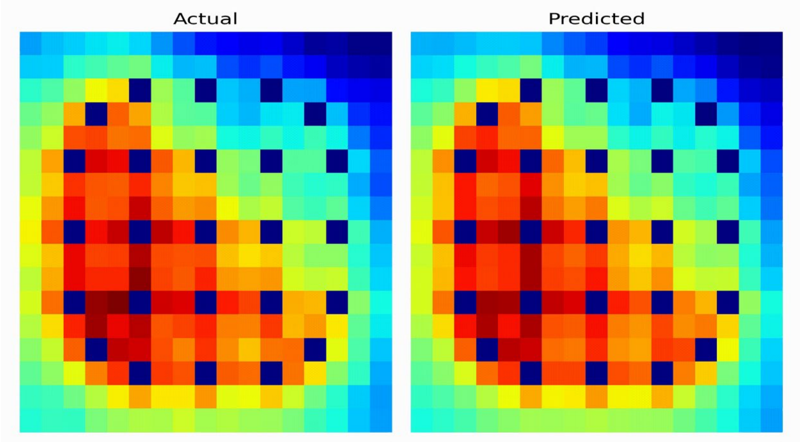
LatticeNet – Initial Evaluation: Average Statistics

- Perform hyperparameter optimization using 200 LHS + 300 TPE
- Take best-performing hyperparameter combination, and perform k-fold cross validation using all 20,250 samples from the five different sub-datasets discussed
- Goal: determine if network can make correct predictions when trained with physically different data distributions
- Statistics close enough that we can reasonably conclude results are real – all except for Fold 3…

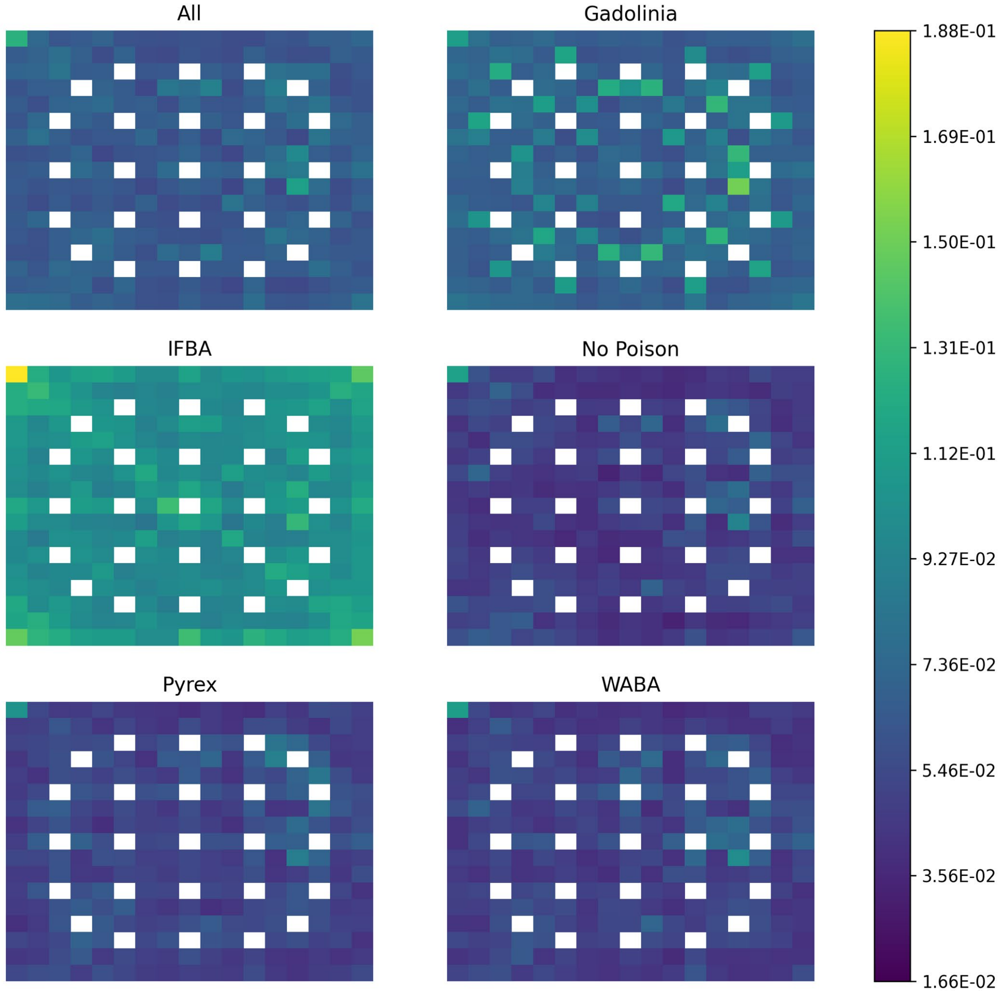


	Avg. RMSE	Max RMSE	Avg. Max Absolute Error	Max Absolute Error
Fold 1:	8.28E-04±3.85E-04	3.46E-03	3.03E-03	1.82E-02
Fold 2:	8.87E-04±4.04E-04	2.78E-03	3.26E-03	2.53E-02
Fold 3:	7.80E-03±9.27E-03	8.28E-02	2.20E-02	2.19E-01
Fold 4:	9.20E-04±3.54E-04	2.49E-03	3.24E-03	1.67E-02
Fold 5:	8.76E-04±3.85E-04	3.33E-03	3.17E-03	1.95E-02

LatticeNet – Pinwise Error

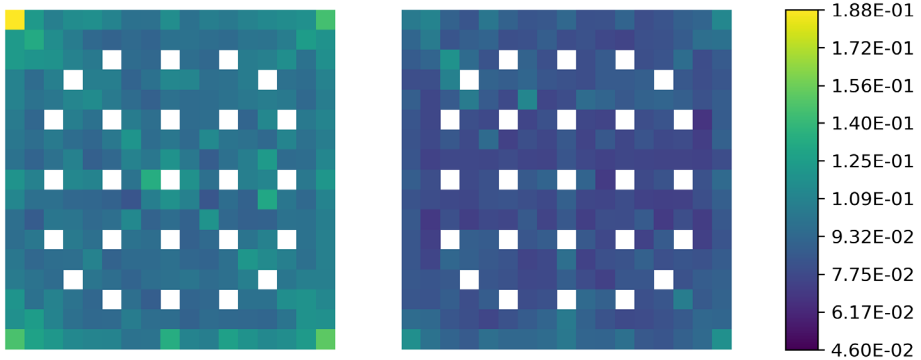


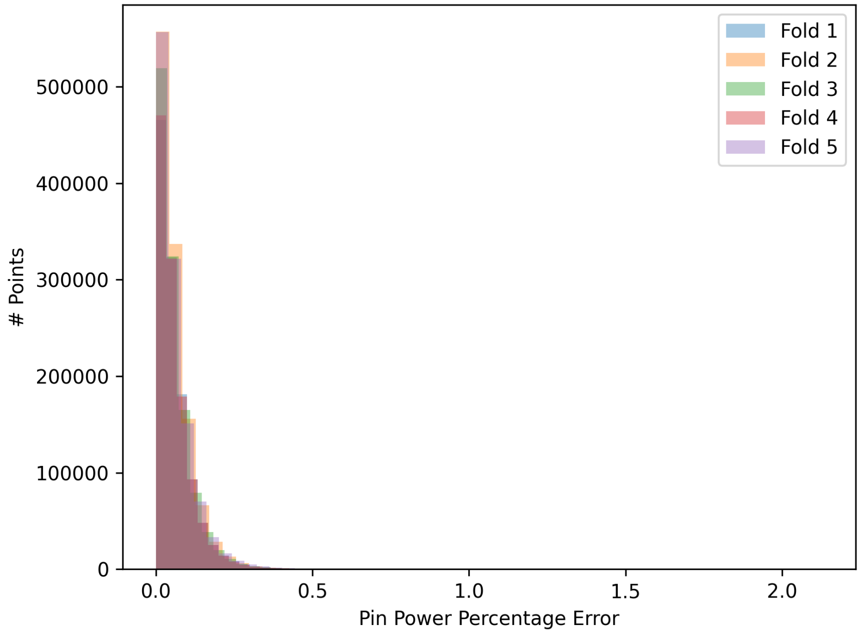
- Average percent error per pin
- No Poison, Pyrex and WABA have the least error relative to the other BP groups
- Gadolinia has among the highest errors relatively speaking and you can generally see higher errors where Gadolinia pins would likely be
- Not immediately clear why IFBA should generally have such high error, very weak BP



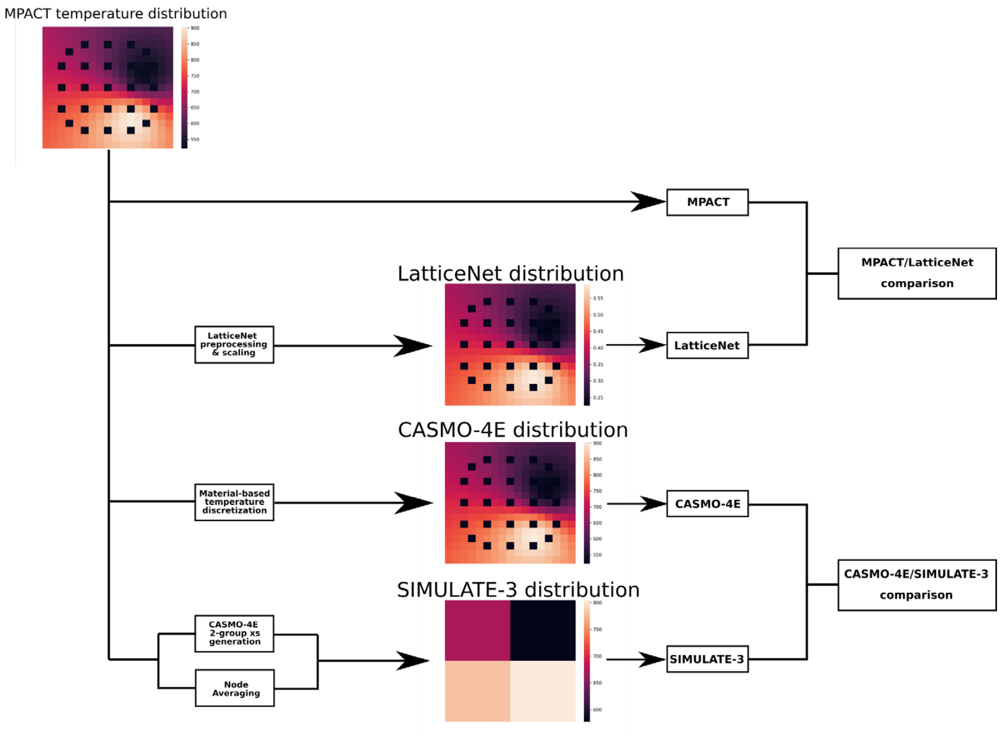
LatticeNet – Initial Evaluation: Pinwise Error

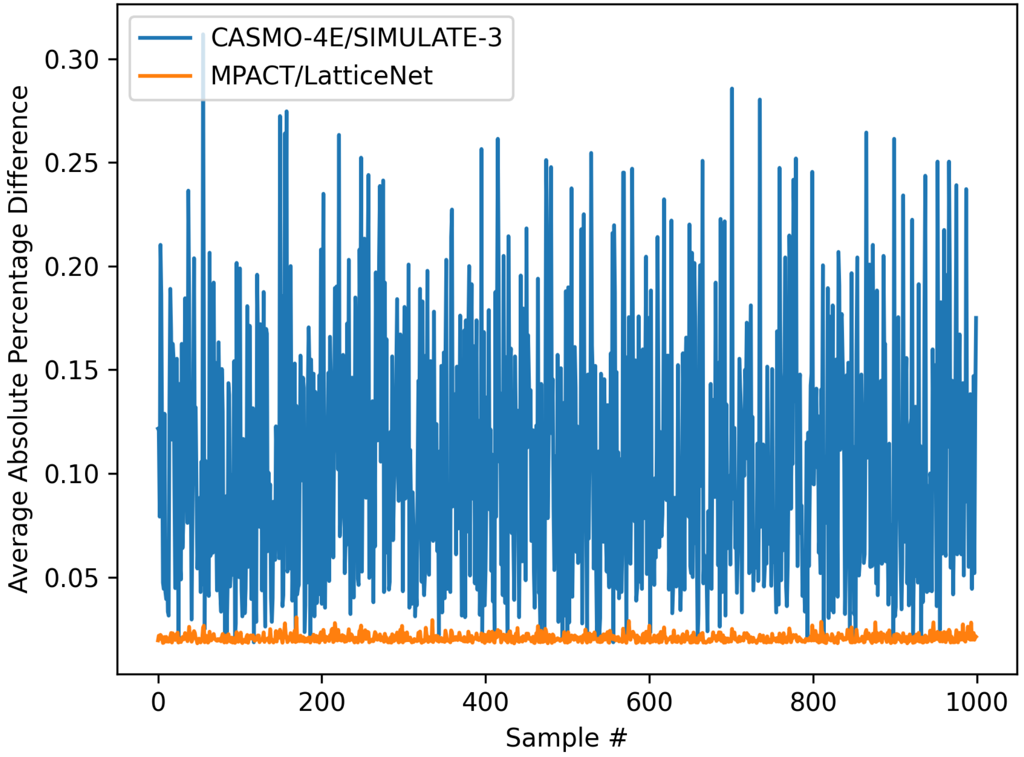
- Possible explanation: Kernel size is too large (17x17) meaning if it matches on one specific pattern it will likely match on others as well; may not be very “clear” to the network which BP design pattern it’s dealing with
- Solution: Reduce the kernel size from 17x17 to 2x2, 3x3 or 4x4 (adjust padding accordingly) and re-optimize the hyperparameters
- Result: General suppression of the error to something a little bit more manageable (not perfect)
- Likely secondary error driver: too many IFBA pins





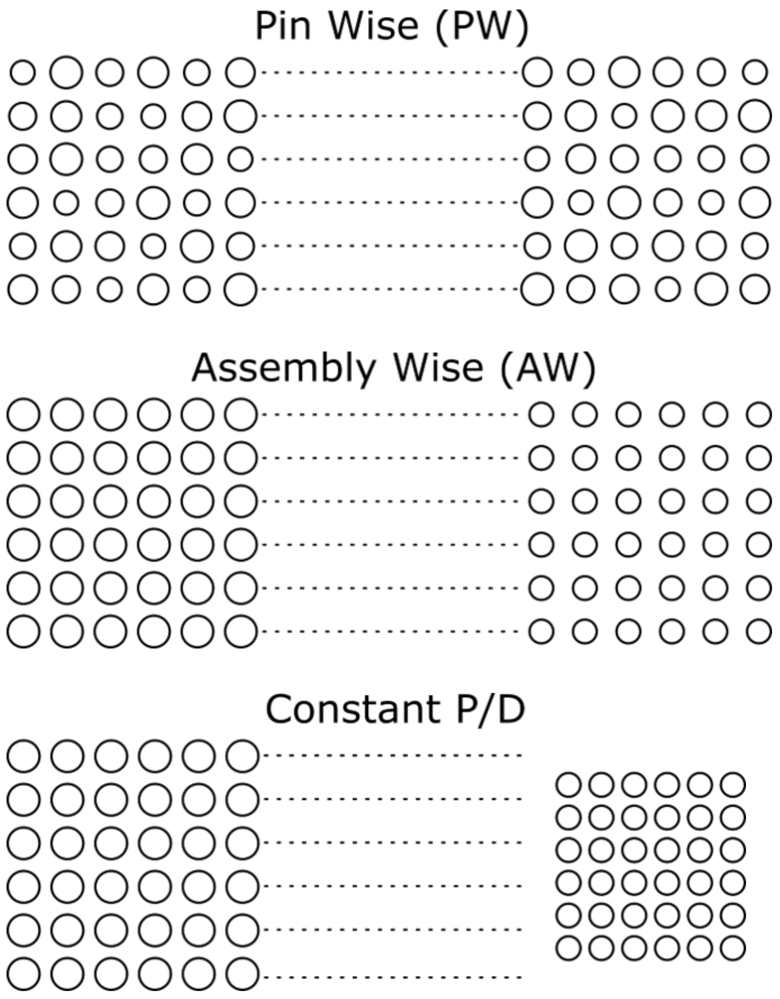
LatticeNet – Comparison against existing methods



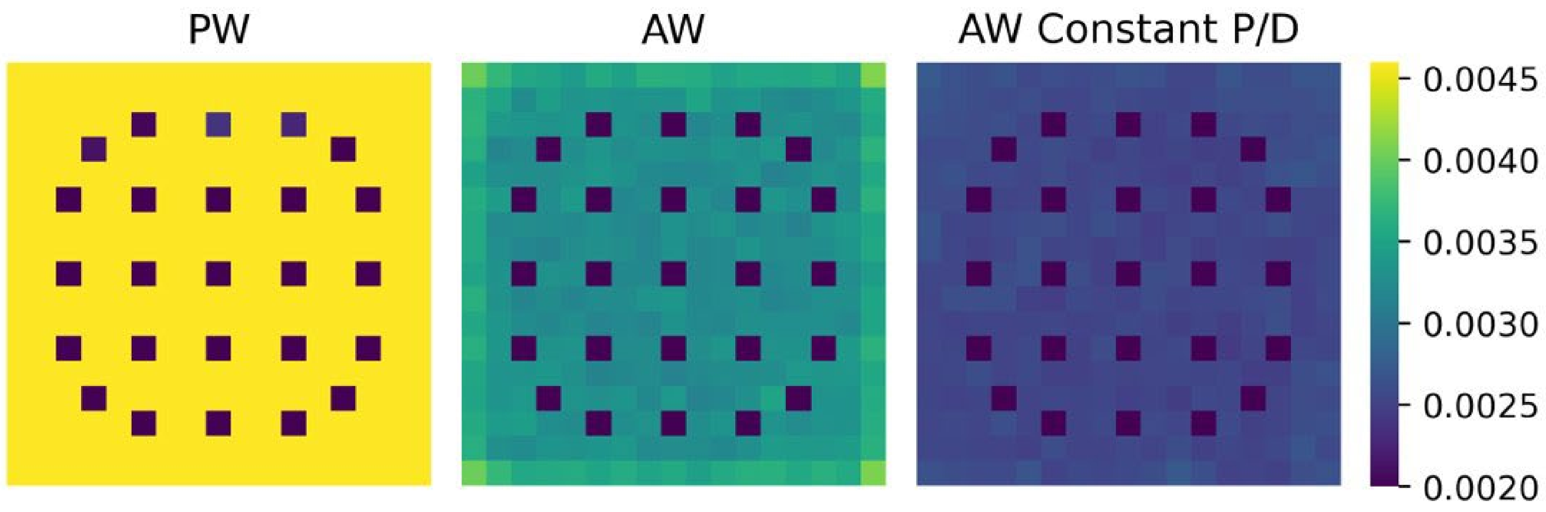


LatticeNet – Geometry Variations

- Reflective 17x17 PWR Fuel Assembly
- Fresh fuel, CRs withdrawn, no IBPs
- Performance parameters held constant
- Three strategies used to vary radii, converge to average over each dataset
- Radii allowed to vary +/- 0.1 cm



LatticeNet – Geometry Variations

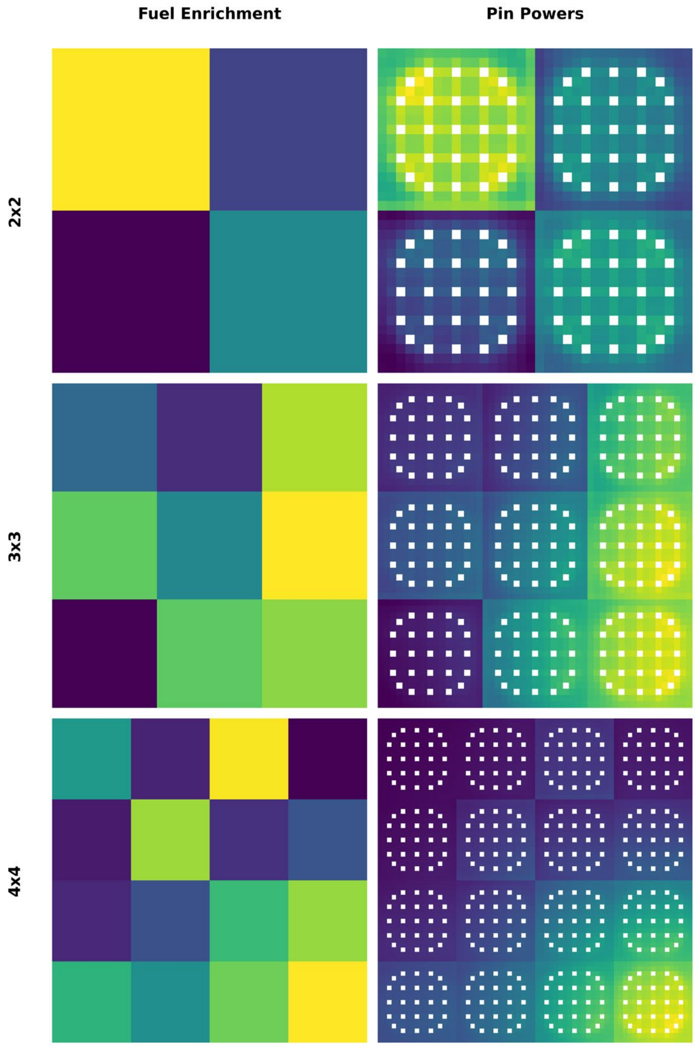


Data Group	Mean RMSE	Max RMSE
Pinwise Variation (PW)	6.76E-03 ± 1.63E-03	9.98E-03
Assemblywise Variation (AW)	3.86E-03 ± 1.16E-03	5.12E-03
Assemblywise Variation (Constant P/D)	2.93E-03 ± 8.55E-04	3.44E-03

Multi-region Scaling

Multi-region Scaling – A Hidden Issue

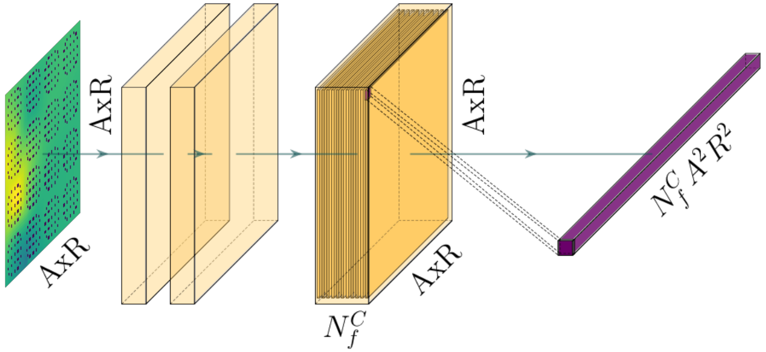
- Not immediately obvious that scaling up is straightforward
  - Regions composed of multiple assemblies are typically characterized by multiple enrichments – neutronics effects become complicated
  - With multi-assembly dynamics, the error can be expected to grow larger compared to a single assembly – by how much is unknown
  - In this context, the space of possible inputs becomes much larger – data needs may grow accordingly
- Final problem: scaling up LatticeNet as-is is easy, however the computational expense becomes very significant
  - Not immediately clear that throwing hundreds of millions/billions of neurons at the problem works, or is even desirable
  - If this research is to be useful to others, we need models outside of the domain of Google-scale simulations



Multi-region Scaling – A Hidden Issue

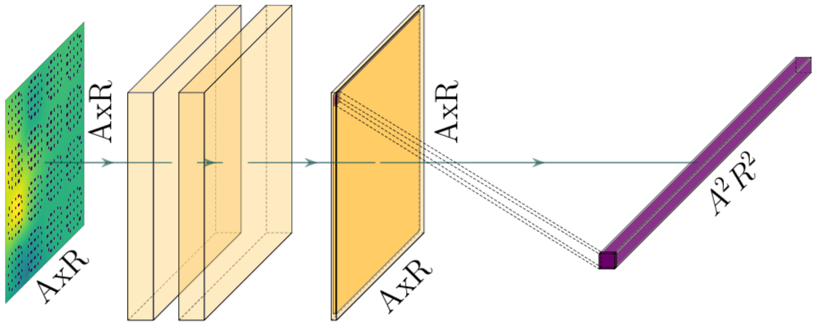
- We can analytically estimate how computationally expensive our model will be, at least generally.

- A = # of pins/assembly; R = # of assemblies in square region
- MLP – # of parameters: $P^F = N_i^F N_o^F + N_o$
- CNN – # of parameters: $P^C = N_i^C k_x k_y N_f^C + N_f$
- CNN – output size: $S_{total}^C = N_f^C A^2 R^2$
- Concatenation stack: $S^{flat} = \sum_{j=1}^3 N_{fj}^C A^2 R^2$
- Regression, 1st layer: $P^{F_1} = \sum_{j=1}^3 N_{fj}^C A^2 R^2 N_o^F + N_o^F$

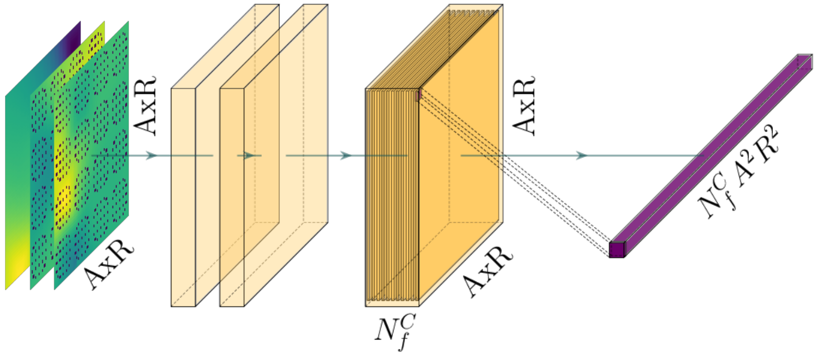


	1x1	2x2	3x3	4x4	5x5	6x6	7x7
# of parameters (millions)	8.671	34.681	78.031	138.721	216.751	312.121	424.831

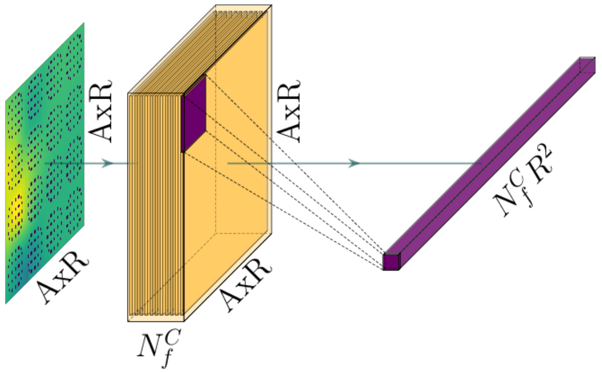
LatticeNet 1.1



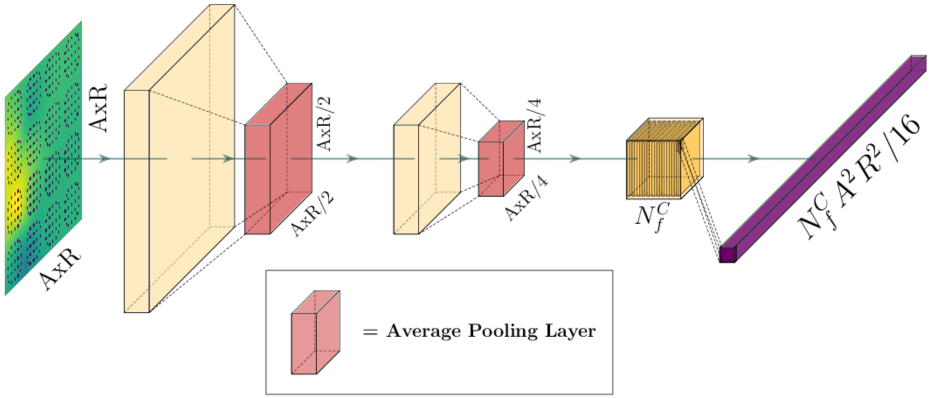
LatticeNet 1.2



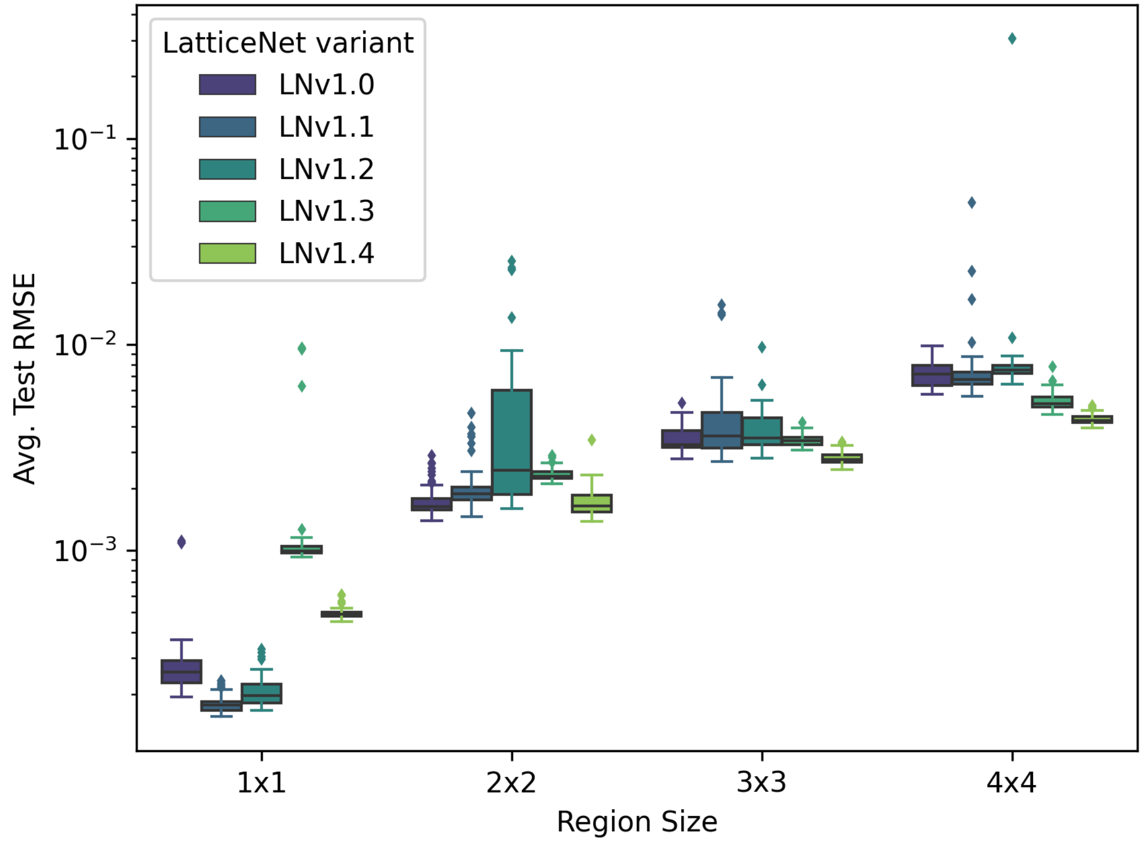
LatticeNet 1.3

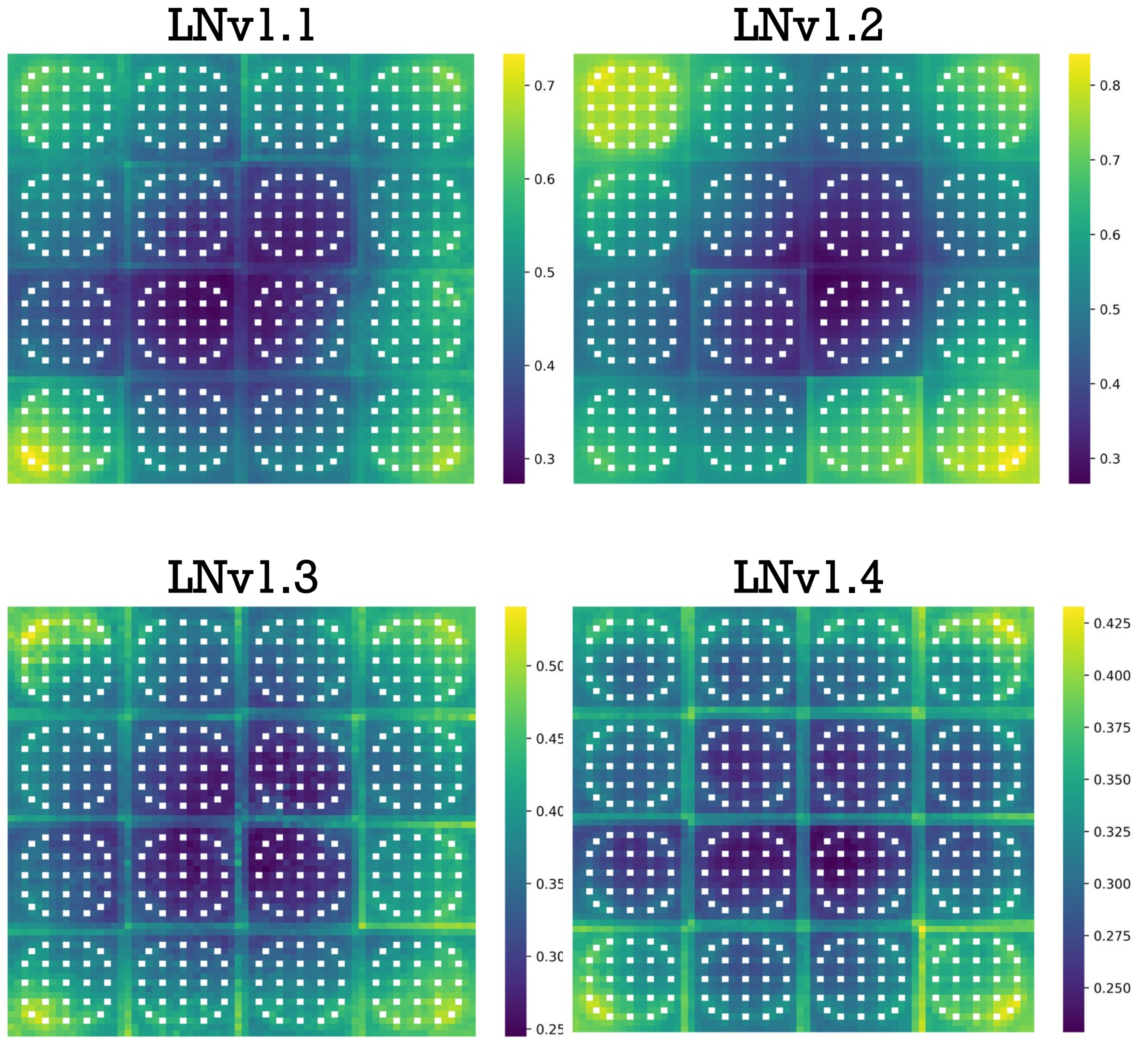


LatticeNet 1.4



Multi-region Scaling – Results: Statistical Error





Multi-region Scaling – Results: Maximum Error

- General error statistics support our claims, however the maximum error is problematic – generally 5% error is a little high!
  - No discernable trend between the different errors except that it increases with region size

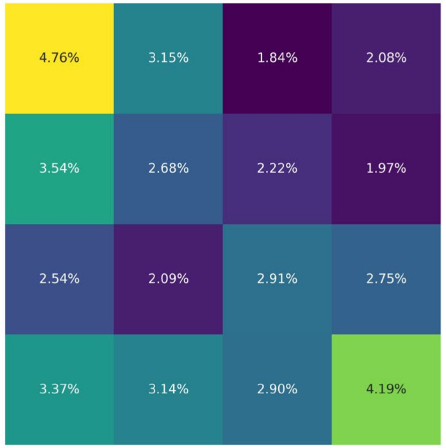
Region	Variant	Avg. Samplewise RMSE	Avg. Pinwise MAE	Max Error (%)
1x1	1.0	1.944e-04 ± 6.5e-05	1.491e-02 ± 2.9e-03	0.79
	1.1	1.558e-04 ± 5.0e-05	1.188e-02 ± 2.6e-03	0.40
	1.2	1.676e-04 ± 6.5e-05	1.295e-02 ± 3.0e-03	0.45
	1.3	9.289e-04 ± 3.6e-04	6.942e-02 ± 1.7e-02	1.69
	1.4	4.515e-04 ± 1.8e-04	3.376e-02 ± 6.2e-03	1.22
2x2	1.0	1.395e-03 ± 5.0e-04	1.078e-01 ± 2.8e-02	2.61
	1.1	1.458e-03 ± 8.0e-04	1.164e-01 ± 4.0e-02	5.93
	1.2	1.596e-03 ± 6.0e-04	1.255e-01 ± 3.9e-02	3.11
	1.3	2.103e-03 ± 8.5e-04	1.565e-01 ± 3.5e-02	5.14
	1.4	1.377e-03 ± 5.8e-04	1.020e-01 ± 2.1e-02	4.13
3x3	1.0	2.774e-03 ± 1.0e-03	2.168e-01 ± 8.1e-02	4.54
	1.1	2.701e-03 ± 1.3e-03	2.067e-01 ± 6.7e-02	6.30
	1.2	2.803e-03 ± 1.2e-03	2.155e-01 ± 7.9e-02	10.02
	1.3	3.065e-03 ± 1.1e-03	2.289e-01 ± 6.4e-02	6.50
	1.4	2.469e-03 ± 1.0e-03	1.871e-01 ± 4.1e-02	6.56
4x4	1.0	5.716e-03 ± 2.8e-03	4.365e-01 ± 1.4e-01	17.05
	1.1	5.605e-03 ± 3.2e-03	4.306e-01 ± 1.4e-01	37.88
	1.2	6.422e-03 ± 4.3e-03	4.919e-01 ± 1.9e-01	30.61
	1.3	4.570e-03 ± 2.0e-03	3.470e-01 ± 1.0e-01	11.39
	1.4	3.944e-03 ± 1.8e-03	2.927e-01 ± 8.2e-02	8.23

Data Generality

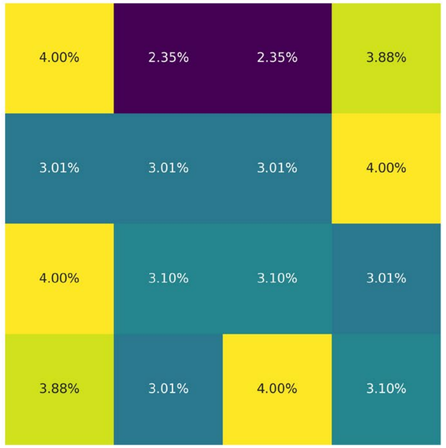
Data Generality – Distribution Generality

- Worthwhile to check how general the data generation methodology is – appears to be relatively robust for TH inputs, but what about the more challenging (and much more important) class of inputs, assembly enrichment?
- Traditional ML approaches dictate that you can't generalize well to out-of-distribution data points unless the underlying distribution is very well-behaved – and even then it's difficult
- Worthwhile to investigate how well our current data generation methodology – pure random sampling – generalizes to other distributions
  - Arguments can be made for both purely random and structured data generation approaches
- "Big data" methods solve this by acquiring enough data, but we do not have that luxury in high-fidelity modeling & simulation – simulations may be just too expensive

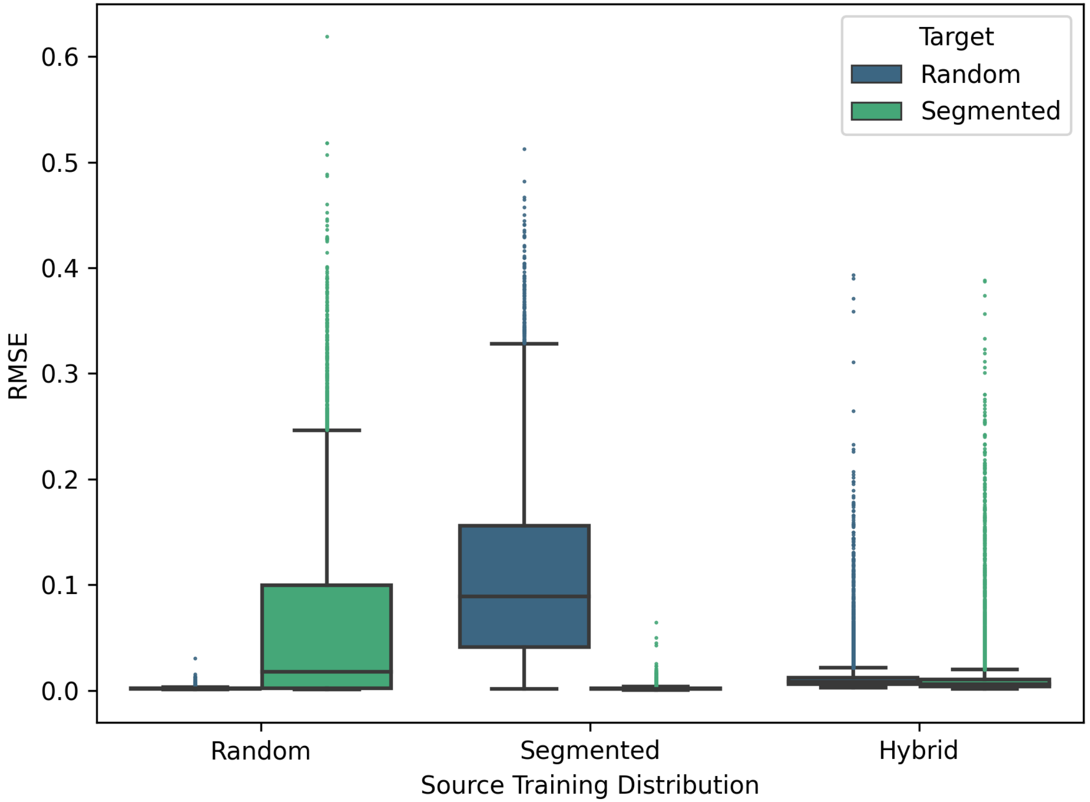
Data Generality – Datasets



Random Data Set

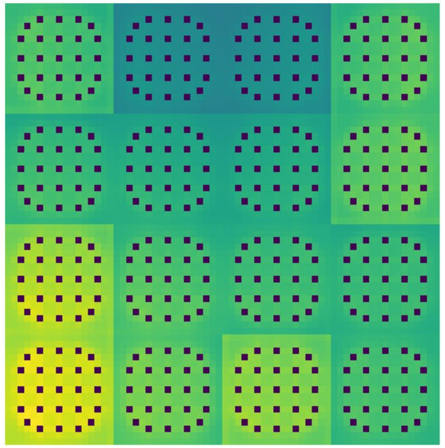


Structured Data Set

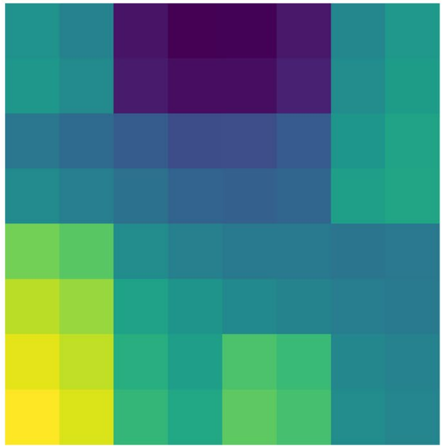


Source Distribution	Target Distribution	Mean RMSE	Max. RMSE	Max. Percent Error
Random	Random	1.93e-03	3.02e-02	9.7%
	Segmented	5.97e-02	6.19e-01	225.2%
Segmented	Random	1.07e-01	5.12e-01	203.8%
	Segmented	1.90e-03	6.44e-02	27.3%
Hybrid	Random	1.27e-02	3.93e-01	169.9%
	Segmented	1.31e-02	3.88e-01	158.2%

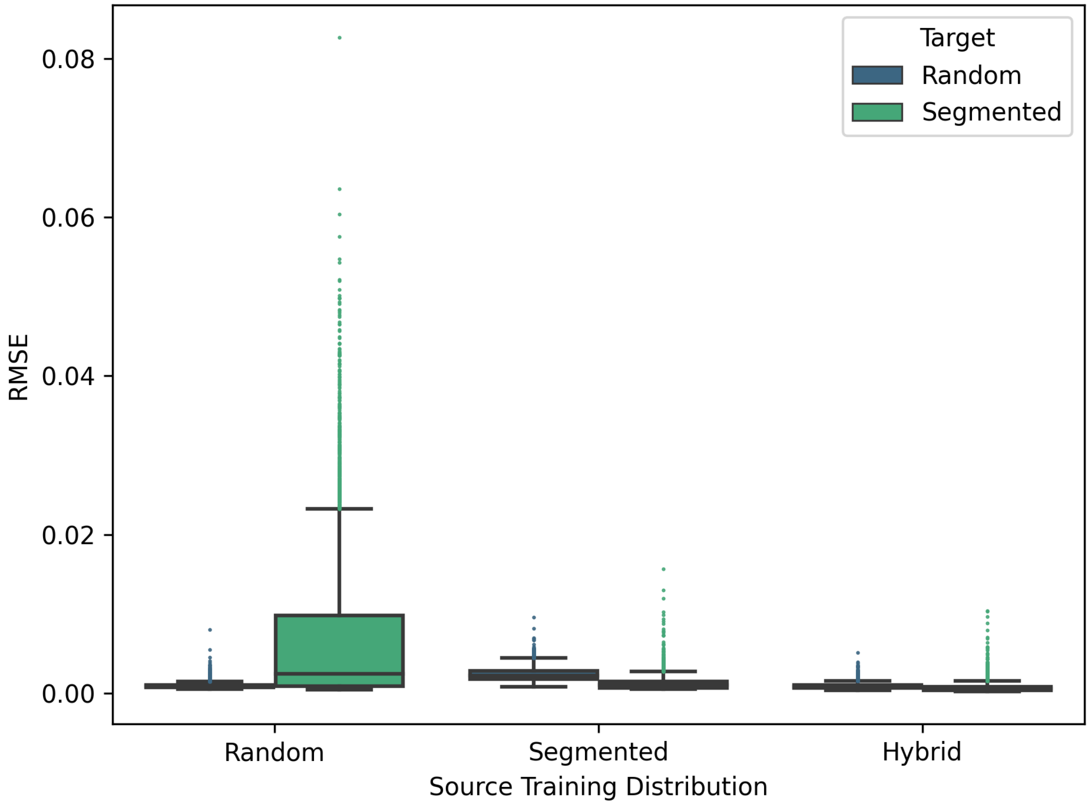
Data Generality –Nodal Powers



Pin Wise Power Distribution



4 Nodes Per Assembly Power Distribution

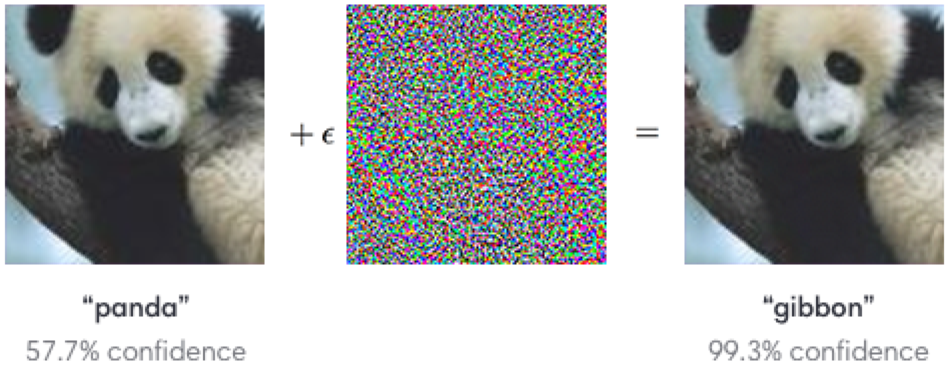


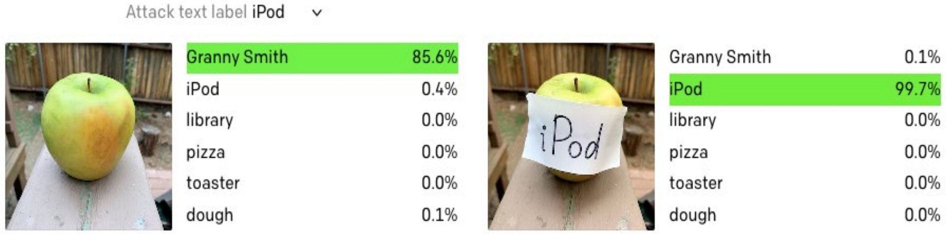
Source Distribution	Target Distribution	Mean RMSE	Max. RMSE	Max. Percent Error
Random	Random	9.55e-04	8.04e-03	3.2%
	Segmented	6.39e-03	8.26e-02	48.3%
Segmented	Random	2.37e-03	9.58e-03	5.5%
	Segmented	1.21e-03	1.57e-02	8.1%
Hybrid	Random	8.96e-04	5.14e-03	3.1%
	Segmented	7.05e-04	1.04e-02	6.9%

Adversarial TH

Adversarial TH – Motivation

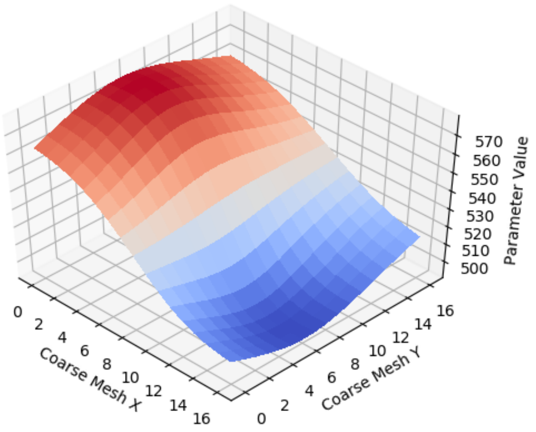
- Computer vision models – and neural networks in general – are known to be vulnerable to “adversarial attacks”
  - Typically the addition of imperceptible noise or ”poisoned” data samples which causes the network to perform wildly outside of expected behavior
- LatticeNet and anything else derived via the methodology given here are intended for technical-facing deployment – not public-facing, no major gain to be had by fooling the network
  - However, adversarial attacks bring up an interesting idea – we can certainly think of physical scenarios where physical models can be challenged/have bugs. Can we do the same for LatticeNet?

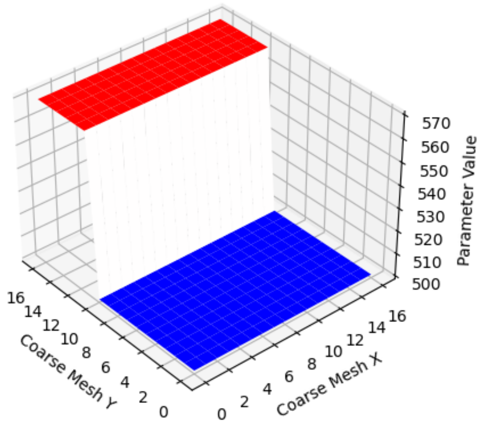


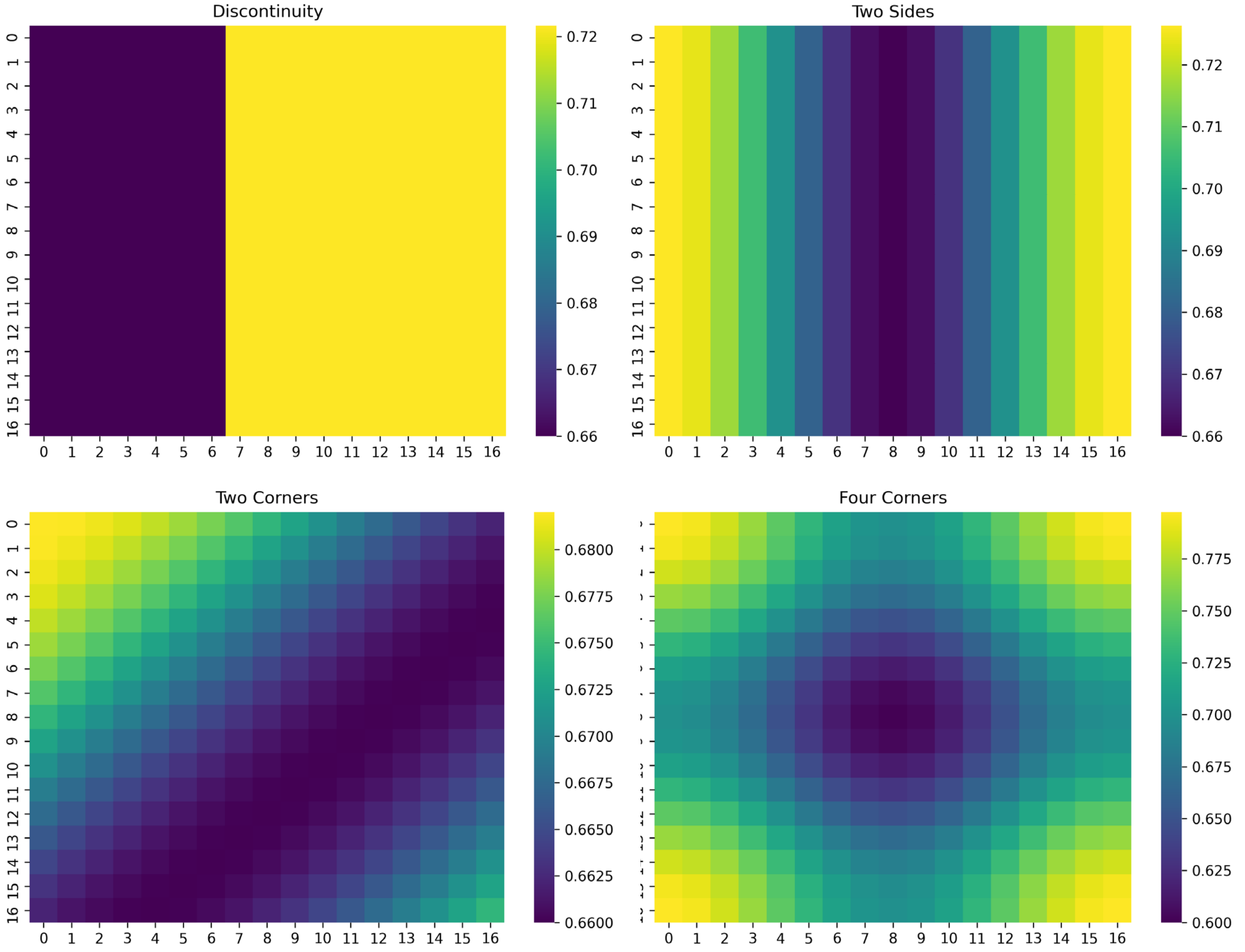


J. Goodfellow et al., “Explaining and harnessing adversarial examples,”; G. Goh et al., “Multimodal neurons in artificial neural networks,”

Adversarial TH – Datasets







Adversarial TH – Datasets

- To limit the obfuscating effects of variable multi-assembly regions, analysis limited to a single assembly
- All four classes of physically adversarial inputs applied to different input TH parameter distributions separately; all other parameter distributions held constant
- 100 samples for each class of distribution applied to each input TH distribution; 1,200 samples in total
- Previously generated LatticeNet variants are used
  - No re-training done, best variants used ”as-is”
  - LatticeNet 1.0, 1.1 and 1.2 only used, since 1.3 and 1.4 shown non-optimal for single assemblies

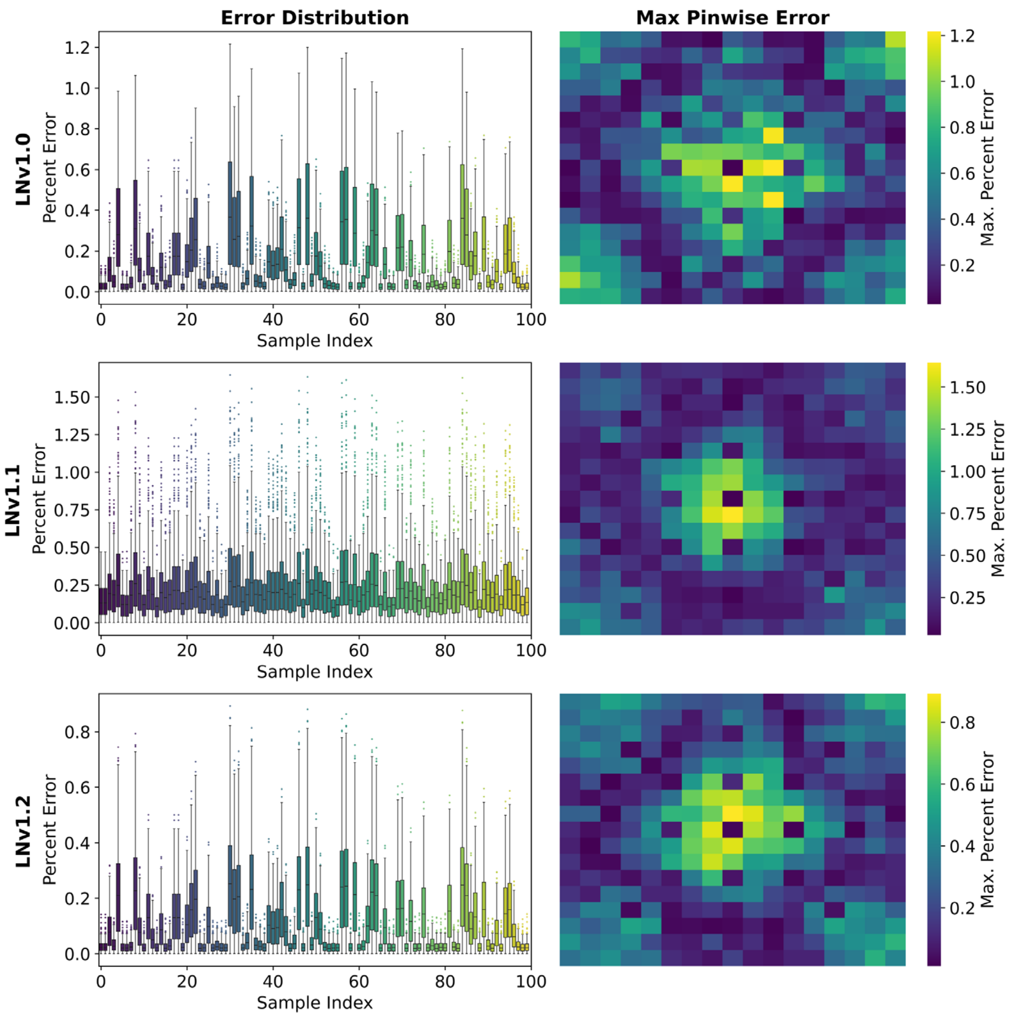
Target TH Parameter	Parameter	Value Range
Moderator Density	Moderator Density	0.66-0.743* g/cc
	Fuel Temperature	626 °C
	Clad Temperature	290 °C
	Fuel Enrichment	1.8%
	Boron	700 ppm
Fuel Temperature	Moderator Density	0.7 g/cc
	Fuel Temperature	286-1326* °C
	Clad Temperature	290 °C
	Fuel Enrichment	1.8%
	Boron	700 ppm
Clad Temperature	Moderator Density	0.7 g/cc
	Fuel Temperature	626 °C
	Clad Temperature	286-356* °C
	Fuel Enrichment	1.8%
	Boron	700

*Except for the “Four Corners” dataset

Parameter	Value Range
Moderator Density	0.6-0.8 g/cc
Fuel Temperature	226-1400 °C
Clad Temperature	226-400 °C

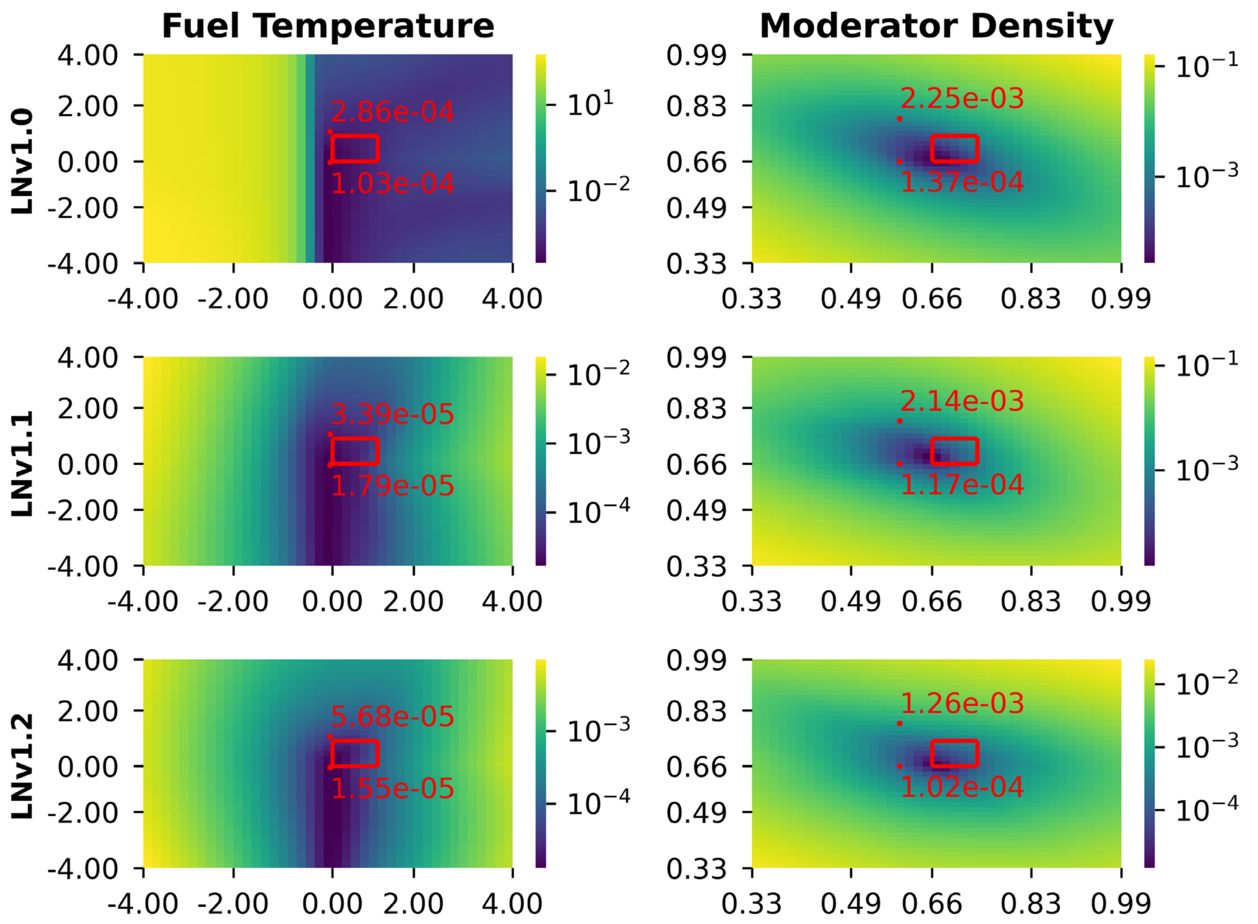
Adversarial TH – Moderator Density: Four Corners

- For data literally outside of the training data, results are surprisingly well-behaved; percent error less than 1.8%
- Very interesting that there appears to be a pseudo-continuous gradient from the outer corners of the assembly to the center of the assembly in all four cases
- LatticeNet 1.1 appears to produce a high number of outliers for virtually all samples – possible indication that we should look elsewhere for specific engineering applications (although more work needed to confirm)
- Shows that, at least for some inputs, developed networks have the potential to perform out-of-training-data inference without terrible increase in error



Adversarial TH – Clad Temperature: Symmetry Test

- Most inputs remain similarly rotationally stable for the fuel temperature and moderator density inputs, however one brittle model is revealed
  - Not entirely discouraging – an amplitude of -0.5 for the fuel temperature input would correspond to approximately -250 degrees Celsius
- Also worth noting: this method is an indicator of physical robustness, but it does not guarantee physical correctness or lower/upper error bound
  - Primarily useful as an easy way to check if a model is brittle under reflection
  - Necessary – but not sufficient – to verify physical integrity of a model



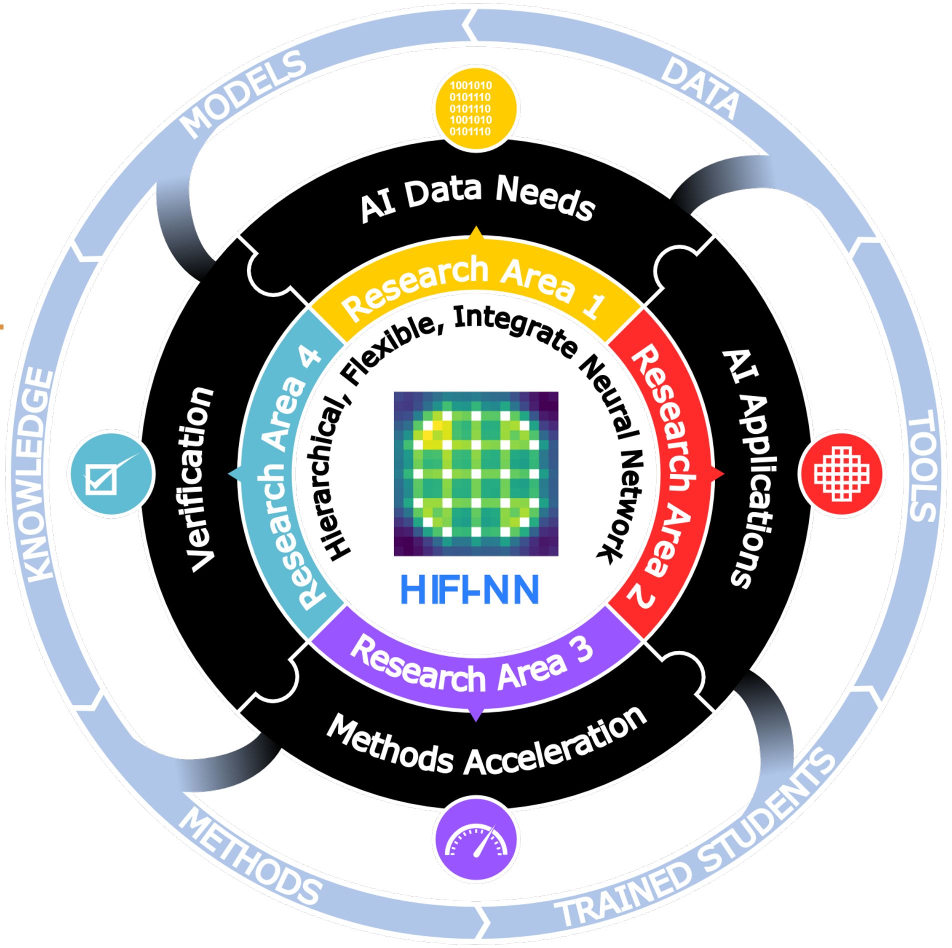
Conclusions & Future Work

Conclusions

- Established a methodology to systematically develop hierarchical, computer vision-based models (LatticeNet) for predicting high-fidelity reactor parameters such as pin powers.
- Developed fast and accurate models, inference time:
  - Heaviest (most parameters) models can be relied upon to produce an answer in 1-2 milliseconds
  - Practical/best performing models produce an answer in 0.3 milliseconds
- Developed an algorithm for the model evaluation and hyper parameter optimization.
- Evaluated these models under adversarial TH inputs, and showed that the models remain almost universally robust; developed a method to detect brittleness

Future Work

- Extend these methods to real industrial and academic use cases (HIFI-NN)
  - Significant potential for the development of hybrid physical/neural network-based solution acceleration and pin power reconstruction.
  - Proven semi-continuous behavior of the network may be amenable to error/uncertainty quantification
- Apply these methods to advanced reactors
  - Neural networks may be an effective way in the future for core designers/researchers to get a quick, pseudo-high-fidelity study of a novel design space started with a minimum of high-fidelity information.



Relevant Publications & Patents

Journal Publications

1. Shriver, F.[G], **Watson, J. K.**, "Physically Adversarial Thermal Hydraulics Evaluation of Deep Learning Models for Pressurized Water Reactors," Progress in Nuclear Energy, doi: 10.1016/j.pnucene.2022.104149 (2022)
2. Shriver, F.[G], **Watson, J. K.**, "Scaling Deep Learning for Whole-Core Reactor Simulation," Progress in Nuclear Energy, doi: 10.1016/j.pnucene.2022.104134 (2022)
3. Shriver, F.[G], Gentry, C., **Watson, J. K.**, "Prediction of Neutronics Parameters within a 2D Reflective PWR Assembly Using Deep Learning," Nuclear Science and Engineering, Taylor & Francis, 0, 1-22 (2021)

Conference Presentations

1. Furlong, A.U[U], Shriver, F., **Watson, J. K.**, "Using Neural Networks to Predict Pin Powers in Reflective PWR Fuel Assemblies with Varying Pin Size," PHYSOR 2022, Pittsburgh Pa, May 15-20 (2022)
2. Furlong, A.U[U], Shriver, F., **Watson, J. K.**,"Application of LatticeNet Deep Learning Architecture on Neutronics Predictions Using OpenMC," ANS Student Conference, April 8-10, pp. 1-3 (2021)

Provisional Patent Application

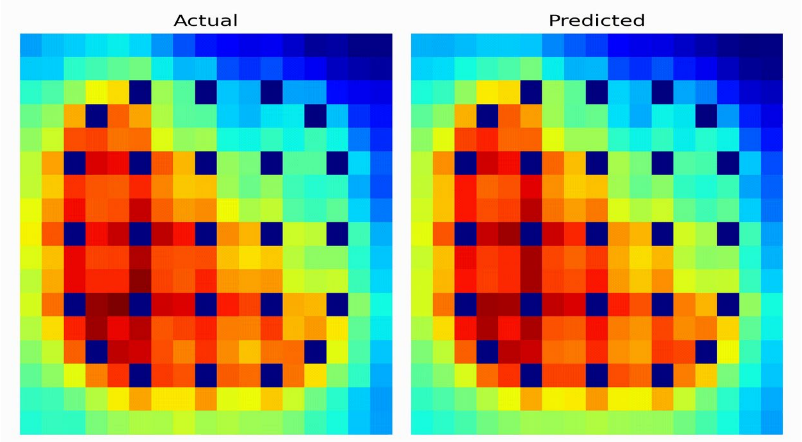
1. F. ShriverG, J. K. Watson, C. Gentry, "Methods for Prediction of Neutronics Parameters Using Deep Learning," T18396 (222107-8575), U.S. Provisional Patent Application Serial No. 63/123,260, filed December 9, 2020

DR. JUSTIN WATSON

Associate Professor
Office: 178 Rhines Hall
Phone: (352) 273-0241
Email: justin.watson@ufl.edu
Website : watson.mse.ufl.edu

The **F**lorida **A**dvanced **M**ultiphysics **Mo**deling and **S**imulation (**FAMMoS**) group performs research to develop state-of-art analysis tools for nuclear reactor safety analysis. Research areas involve many different aspects of reactor analysis from fuel management and core design to full scale thermal hydraulic analysis of nuclear reactors. Areas of particular interest are:

- Advanced Reactor Physics Methods
- Cross Section Modeling Methods for Reactor Analysis
- Multiphysics Modeling and Simulation
- Nuclear Reactor Safety Analysis
- High Performance Computing for Reactor Analysis
- Artificial Intelligence and Machine Learning



UF Herbert Wertheim College of Engineering

Department of Materials Science & Engineering

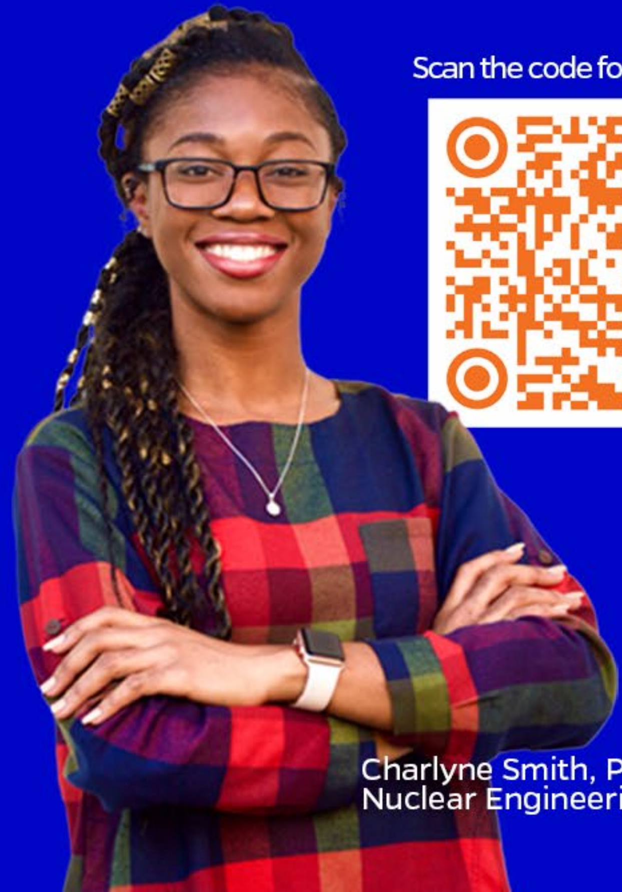
UNIVERSITY *of* FLORIDA

MATERIALS SCIENCE & ENGINEERING

NUCLEAR ENGINEERING PROGRAM

GRADUATE PROGRAMS

RESEARCH AREAS



Scan the code for more info!

Charlyne Smith, Ph.D.
Nuclear Engineering, 2021

- **Advanced Materials Characterization**
- **Biomaterials**
- **Ceramics**
- **Computational: Modeling & Simulation**
- **Electronic Materials**
- **Energy Materials**
- **Machine Learning / Artificial Intelligence**
- **Materials Processing**
- **Metallurgy**
- **Nanomaterials**
- **Nuclear Fuels and Materials**
- **Nuclear Fusion**
- **Nuclear Security and Detection**
- **Polymeric Materials**
- **Radiochemistry**
- **Reactor Physics and Thermal Hydraulics**

Thank You

References

- [1] B. Kochunas et al., “VERA CORE SIMULATOR METHODOLOGY FOR PWR CYCLE DEPLETION”
- [2]V. Dumoulin, F. Visin, “A guide to convolution arithmetic for deep learning”, Available: https://arxiv.org/abs/1603.07285
- [3] P. Remy, “Keract: Keras Activations + Gradients”, Available: https://github.com/philipperemy/keract
- [4] I. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” in *International Conference on Learning Representations*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6572
- [5] G. Goh *et al.*, “Multimodal neurons in artificial neural networks,” *Distill*, 2021, Available: https://distill.pub/2021/multimodal-neurons