SPEED: Switching Power Estimation using Machine Learning with Time-Efficient Data Collection

Yuki Ito and Hoang Nguyen

Table of Contents

Introduction

- Power estimation

$$P_{total} = P_{dynamic} + P_{static}$$

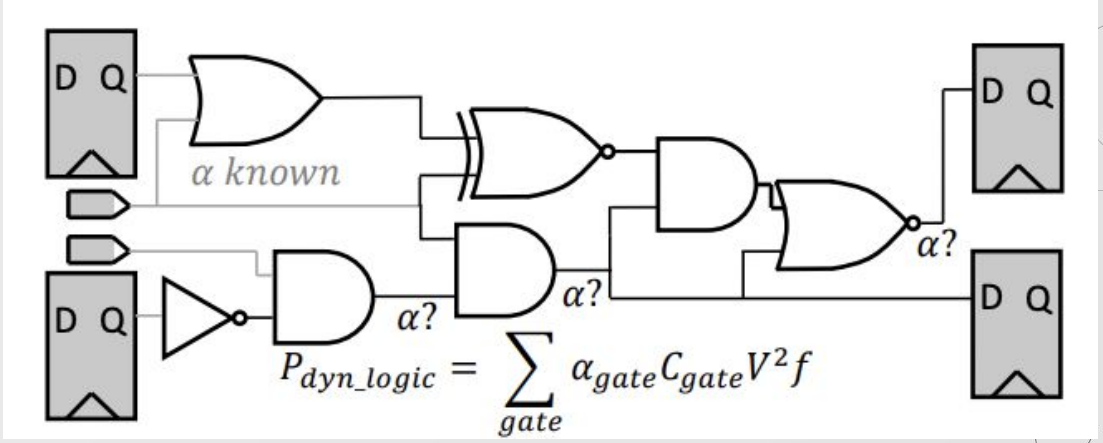
$$P_{dynamic} = P_{switching} + P_{short-circuit}$$

$$P_{switching} = \alpha C V_{DD}^2 f$$ performance-variant

- Simulation time scales with design size
  - Running PAR
  - Separate Power Analysis Tool such as Cadence Voltus

Introduction

- Switching Activity Estimator (SAE)
  - Probabilistic approach to measure toggle rates of each register
  - Foregoes the need to perform cycle-by-cycle power calculation
  - Issues with accuracy, signal correlation or reconvergence
    - Clock domain crossing, etc.



[1] Y. Zhang, H. Ren and B. Khailany, "GRANNITE: Graph Neural Network Inference for Transferable Power Estimation,"

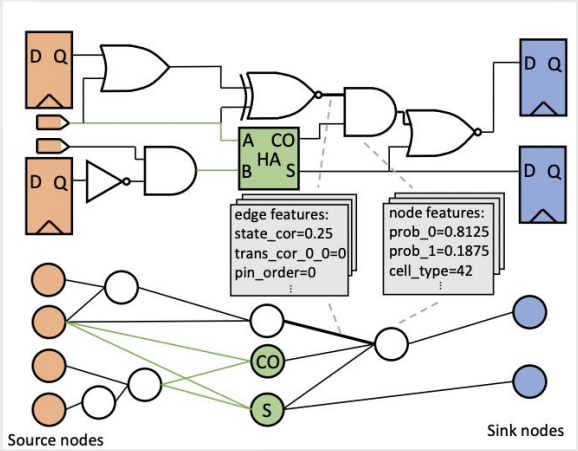
State of the Art SAEs

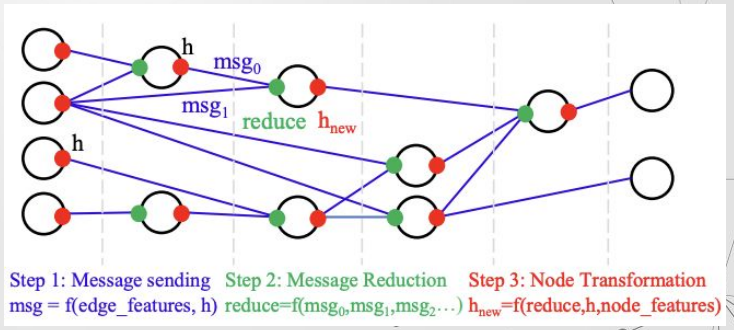
- GRANNITE: Graph Neural Network (GNN) Inference for Transferable Power Estimation (NVIDIA)
    - Captures nonlinear behavior the purely probabilistic SAEs fails to do
        - > 18.7X speedup
        - < 5.5% error rate on benchmarks such as RISC-V Rocket Core (SmallCore), 32-bit full adders, etc.

[1] Y. Zhang, H. Ren and B. Khailany, "GRANNITE: Graph Neural Network Inference for Transferable Power Estimation,"

State of the Art SAEs

- GRANNITE: Graph Neural Network (GNN) Inference for Transferable Power Estimation (NVIDIA)
  - Translate gate-level netlist to graph objects in their Neural Network architecture
    - Given a graph representation of the gate-level netlist, what will be the total switching power with some testbench?

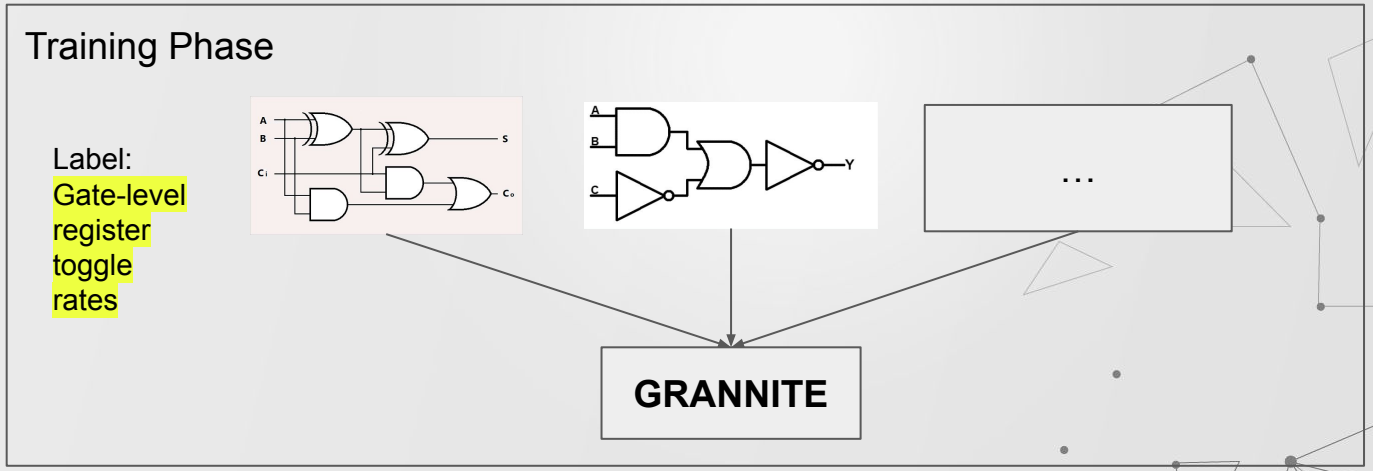




[1] Y. Zhang, H. Ren and B. Khailany, "GRANNITE: Graph Neural Network Inference for Transferable Power Estimation,"

State of the Art SAEs

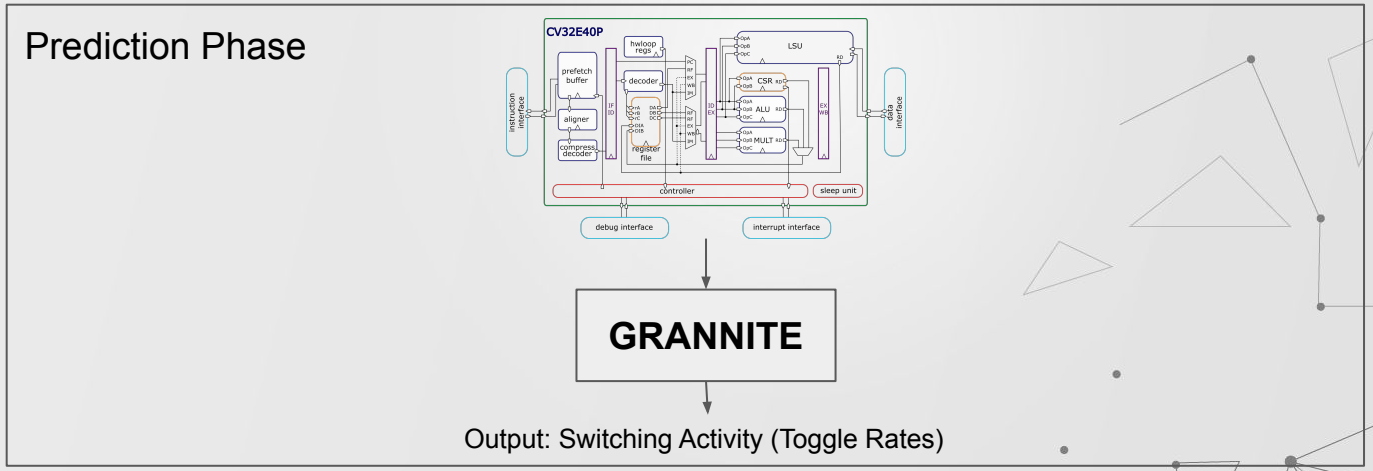
- GRANNITE: Graph Neural Network (GNN) Inference for Transferable Power Estimation (NVIDIA)
  - GPU-accelerated **Transferable** Power Estimator using Graph Convolutional Networks (GCN)



[1] Y. Zhang, H. Ren and B. Khailany, "GRANNITE: Graph Neural Network Inference for Transferable Power Estimation,"

State of the Art SAEs

- GRANNITE: Graph Neural Network (GNN) Inference for Transferable Power Estimation (NVIDIA)
  - GPU-accelerated **Transferable** Power Estimator using Graph Convolutional Networks (GCN)



[1] Y. Zhang, H. Ren and B. Khailany, "GRANNITE: Graph Neural Network Inference for Transferable Power Estimation,"

State of the Art SAEs

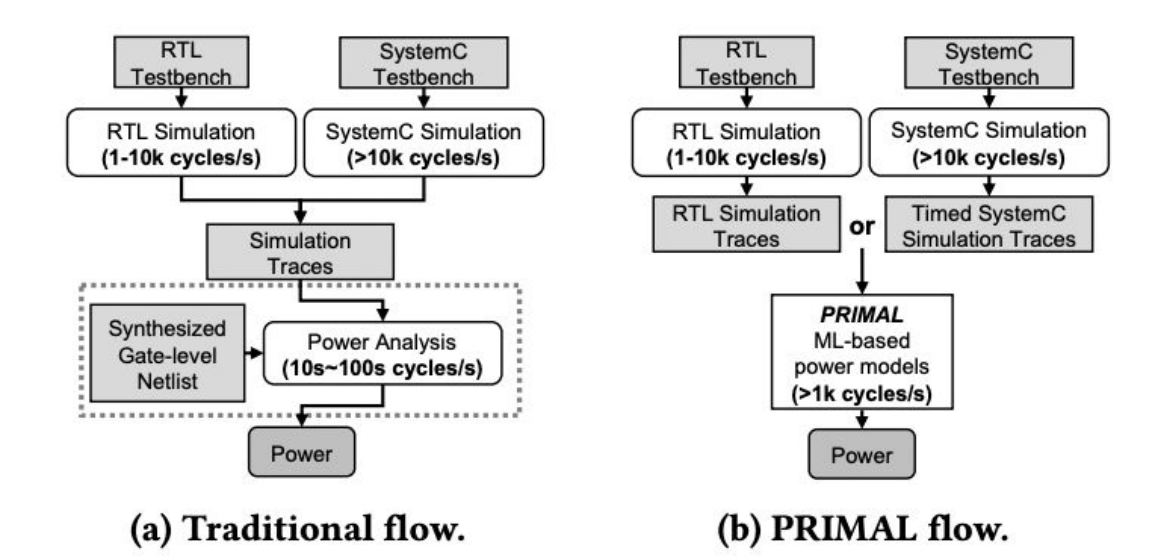
- PRIMAL: Power Inference using Machine Learning (Cornell University, NVIDIA)
    - Cycle-to-cycle gate-level power estimation, non-transferable
    - 15X Speedup

- …..And many more that we won't cover for the interest of time

[2] Y. Zhou, H. Ren, Y. Zhang, B. Keller, B. Khailany and Z. Zhang, "PRIMAL: Power Inference using Machine Learning,"

State of the Art SAEs

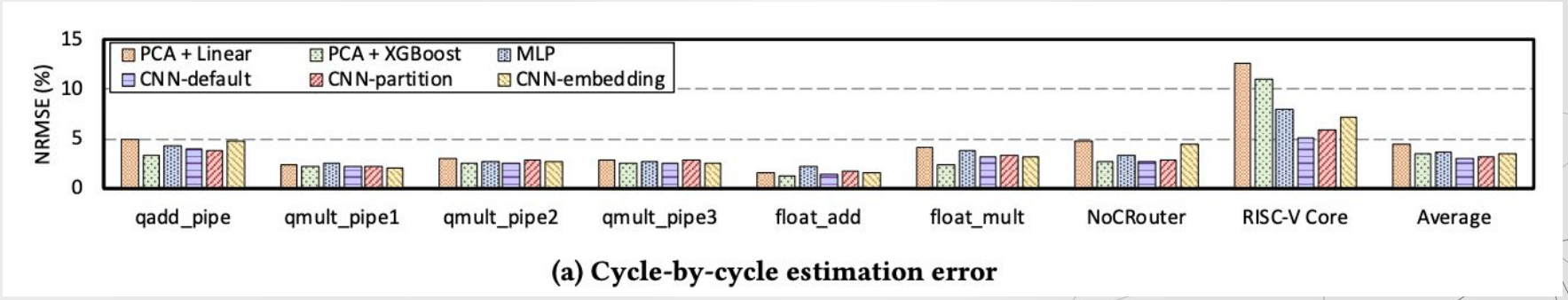
- PRIMAL: PoweR inference using MAchine Learning (Cornell University, NVIDIA)



[2] Y. Zhou, H. Ren, Y. Zhang, B. Keller, B. Khailany and Z. Zhang, "PRIMAL: Power Inference using Machine Learning,"

State of the Art SAEs

- PRIMAL: PoweR inference using MAchine Learning (Cornell University, NVIDIA)

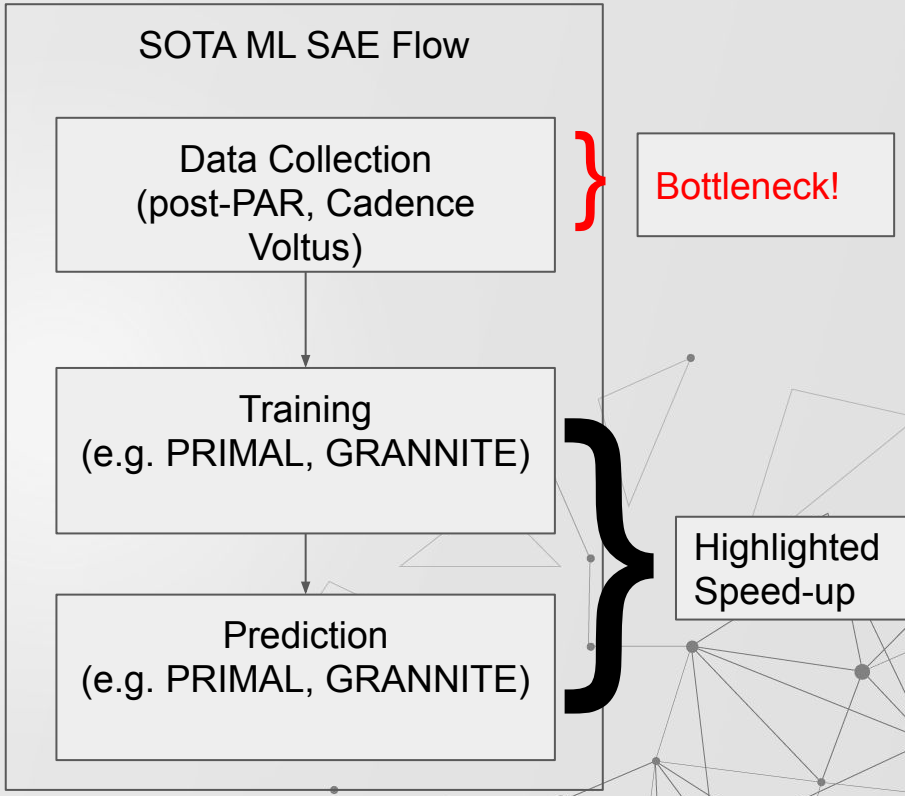


(a) Cycle-by-cycle estimation error

[2] Y. Zhou, H. Ren, Y. Zhang, B. Keller, B. Khailany and Z. Zhang, "PRIMAL: Power Inference using Machine Learning,"

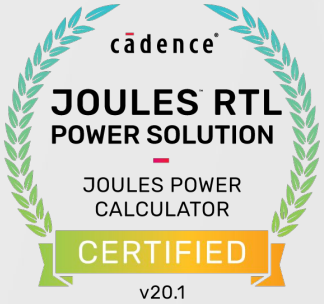
Our Research Project: Motivation

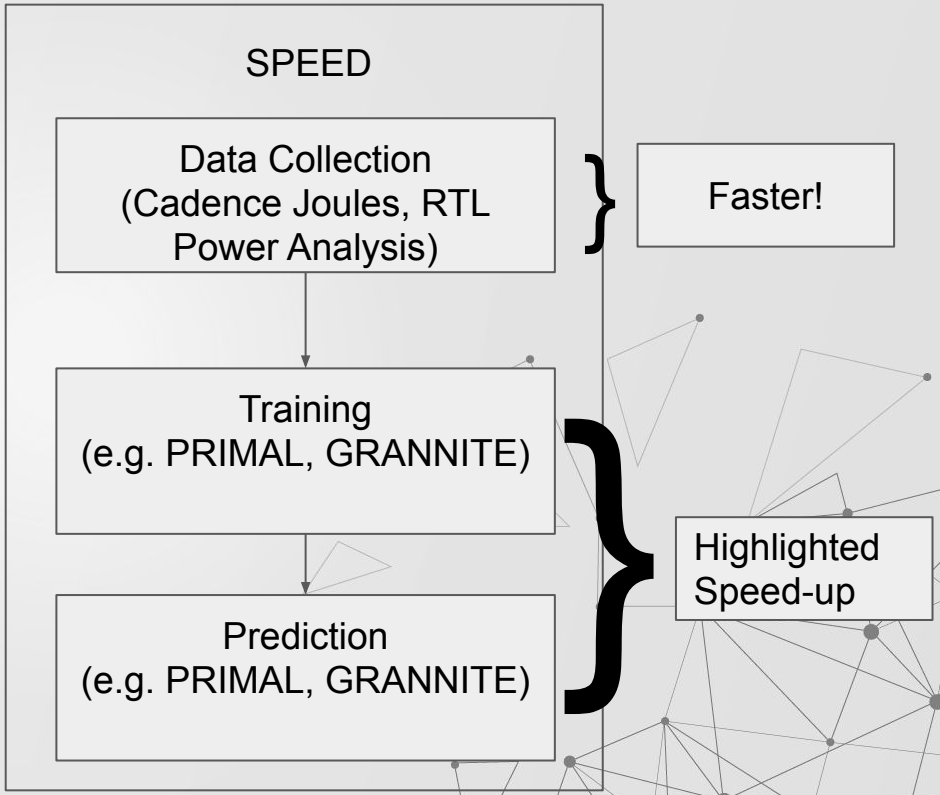
- Data collection is still time consuming
  - SOTA methodology: Supervised Learning
    - Data collection from post-PAR gate-level Power Analysis
    - Infeasible to mimic for a class project
    - Can be applied to project that require quick power estimation
      - unless it is transferable like GRANNITE



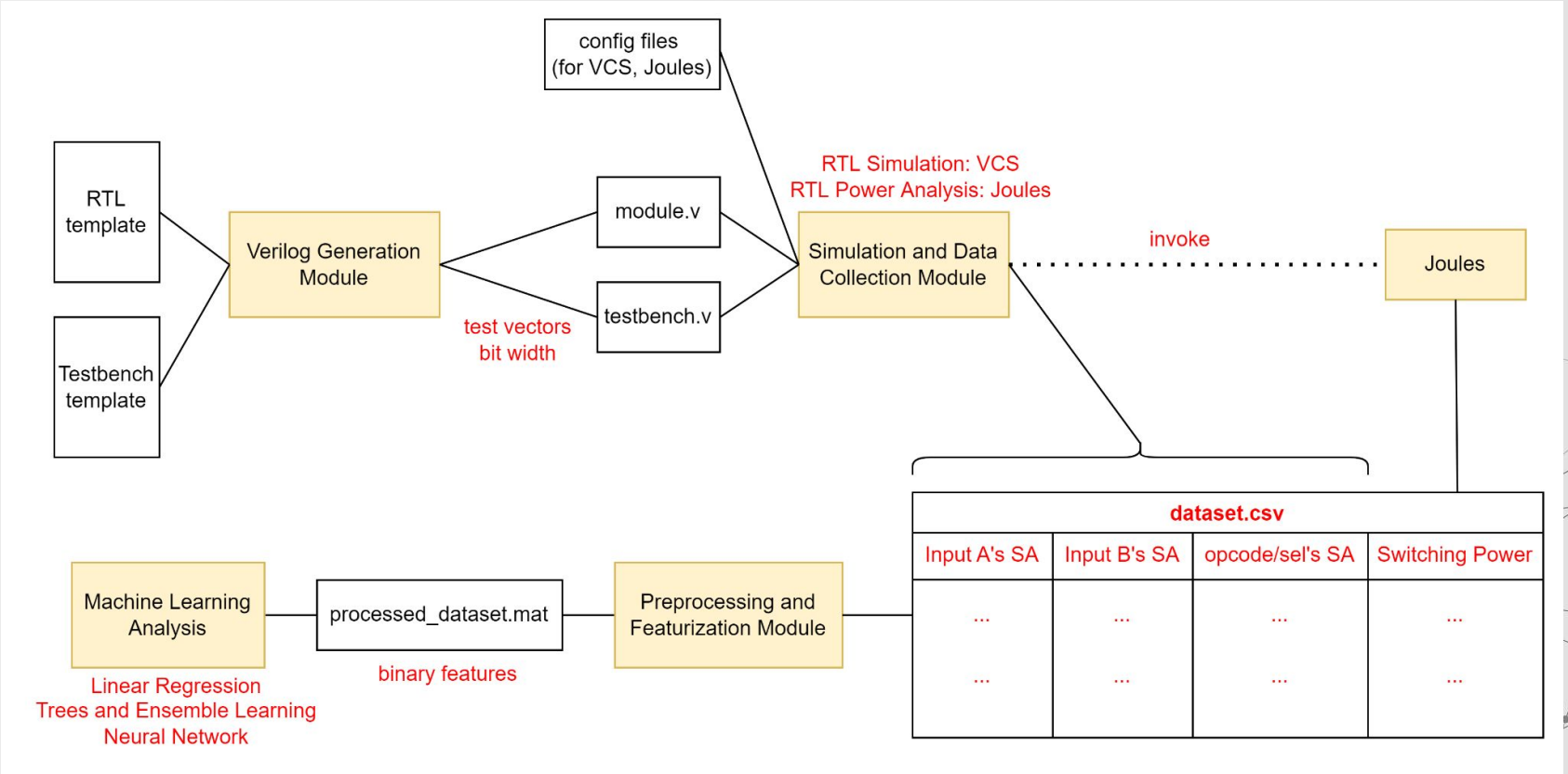
Our Research Project: SPEED

- Fast data collection via RTL Power Estimation
  - As opposed to Gate-level Power Estimation
- Testing ML methods on representative combinational logic circuits



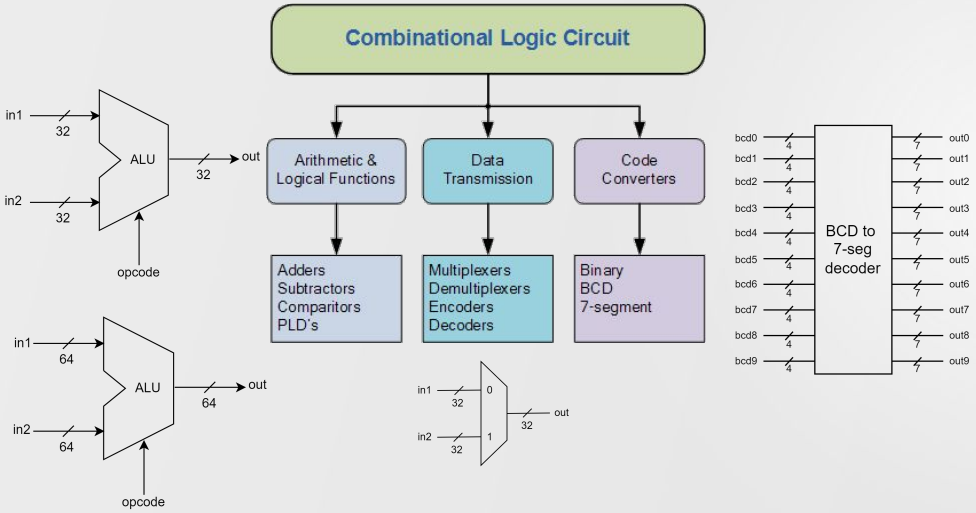


Methodology: Data Collection Pipeline



Methodology: Data Collection

- Choice of combinational logic circuits:

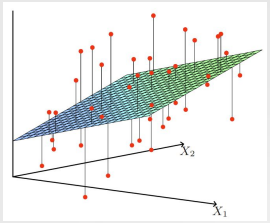


- Collected ~ 5000 samples for each circuit
- Featurization: Using input switching activities
  - 0 if input bit remains the same
  - 1 if input bit switches state

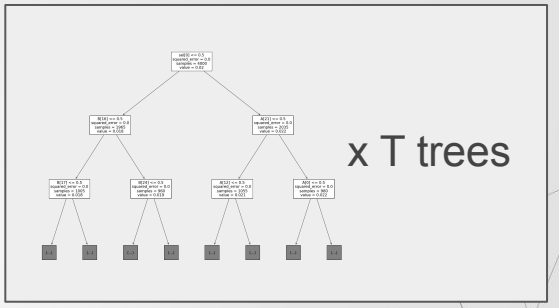
Circuit	Number of Features
ALU32	68
ALU64	132
MUX32	66
7SEG	40

Methodology: Machine Learning Analysis

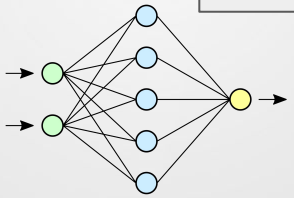
- Linear regression
  - Least Squares
  - Ridge Regression
  - LASSO



- Trees and Ensemble Learning
  - Decision Trees
  - Random Forests
  - AdaBoost for Decision Trees (Boosted Trees)



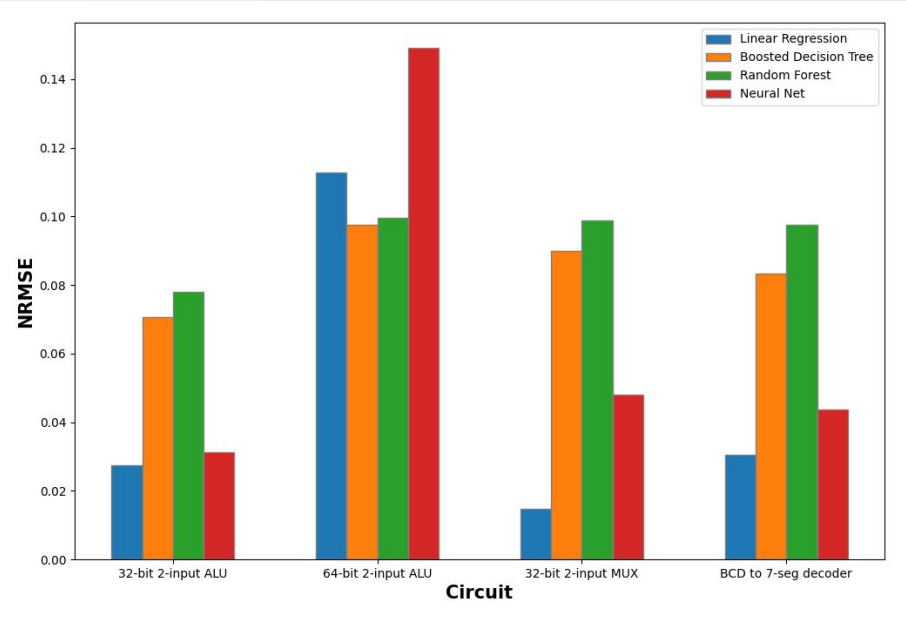
- Neural Network
  - 0, 1, 2, or 3 hidden layers
  - With or without dropout layers



- Evaluation: NRMSE

$$NRMSE = \frac{1}{y_{max} - y_{min}} \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}}$$

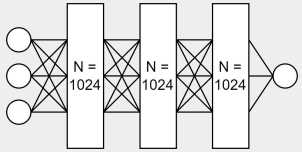
Results



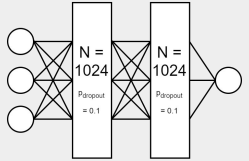
Circuit	ALU32	ALU64	MUX32	7SEG
Best Tree Depth (Boosted Decision Trees)	7	4	5	6
# Trees (Random Forest)	100			

Best Neural Net architectures:

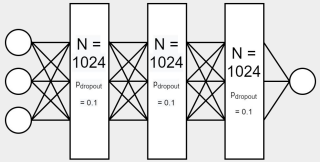
- ALU32:



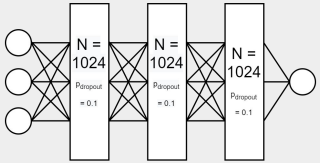
- ALU64:



- MUX32:



- 7SEG:



Limitation/Future Work

1. **Collect post-PAR gate-level power data**

   Instead of RTL-level power data
   (time complexity vs accuracy tradeoff)

2. **Featurize switching activities from intermediate/output gates**

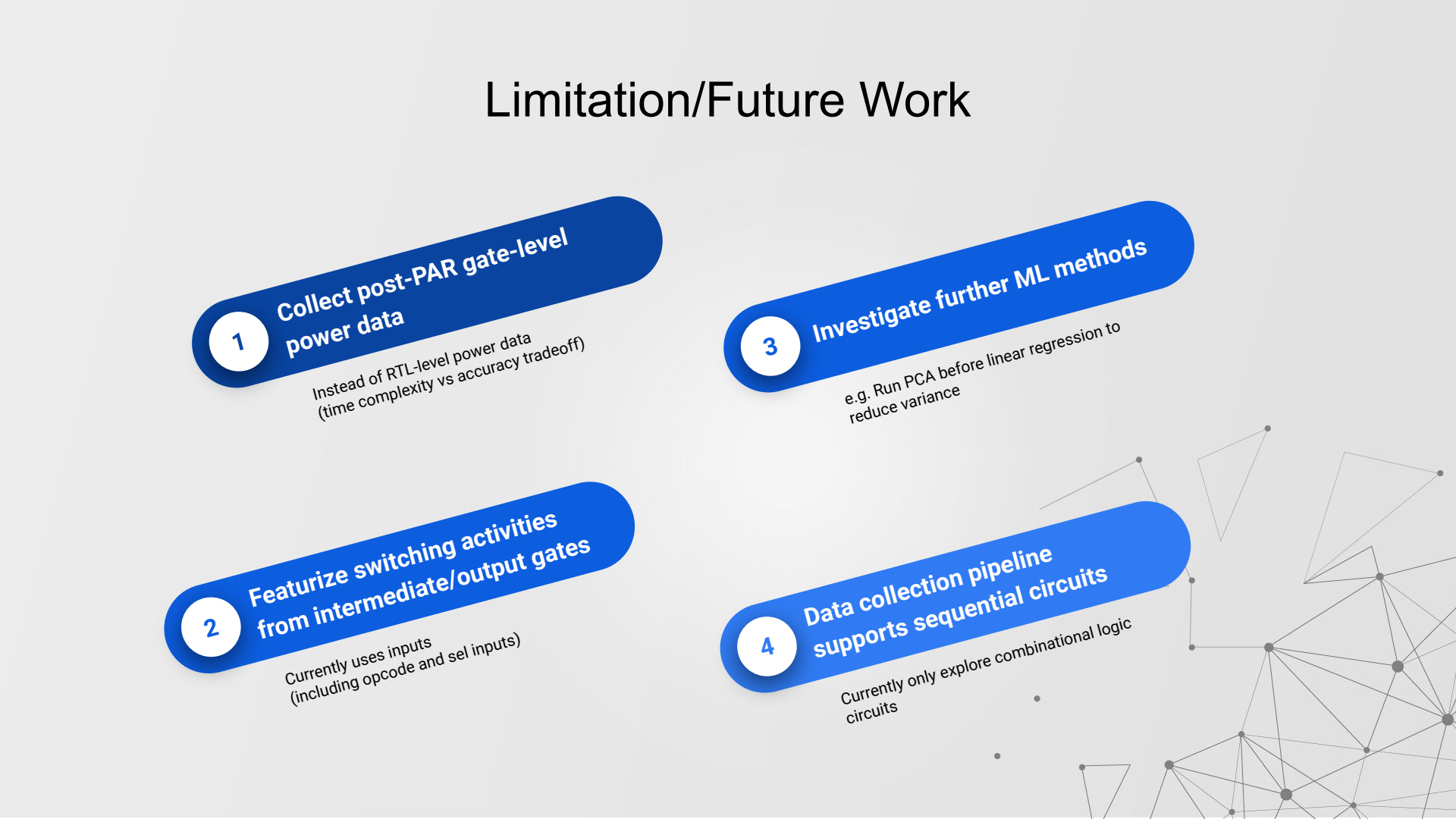
   Currently uses inputs
   (including opcode and sel inputs)

3. **Investigate further ML methods**

   e.g. Run PCA before linear regression to reduce variance

4. **Data collection pipeline supports sequential circuits**

   Currently only explore combinational logic circuits



References

[1] Y. Zhang, H. Ren and B. Khailany, "GRANNITE: Graph Neural Network Inference for Transferable Power Estimation," *2020 57th ACM/IEEE Design Automation Conference (DAC)*, 2020, pp. 1-6, doi: 10.1109/DAC18072.2020.9218643.

[2] Y. Zhou, H. Ren, Y. Zhang, B. Keller, B. Khailany and Z. Zhang, "PRIMAL: Power Inference using Machine Learning," *2019 56th ACM/IEEE Design Automation Conference (DAC)*, 2019, pp. 1-6.