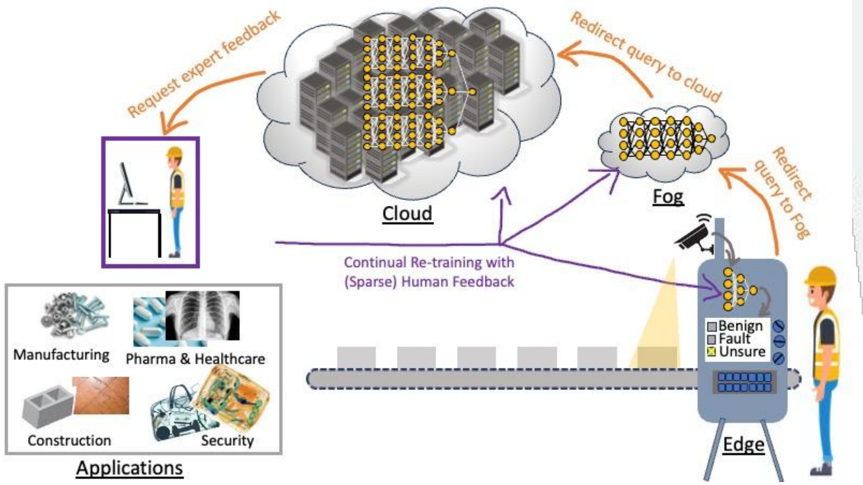
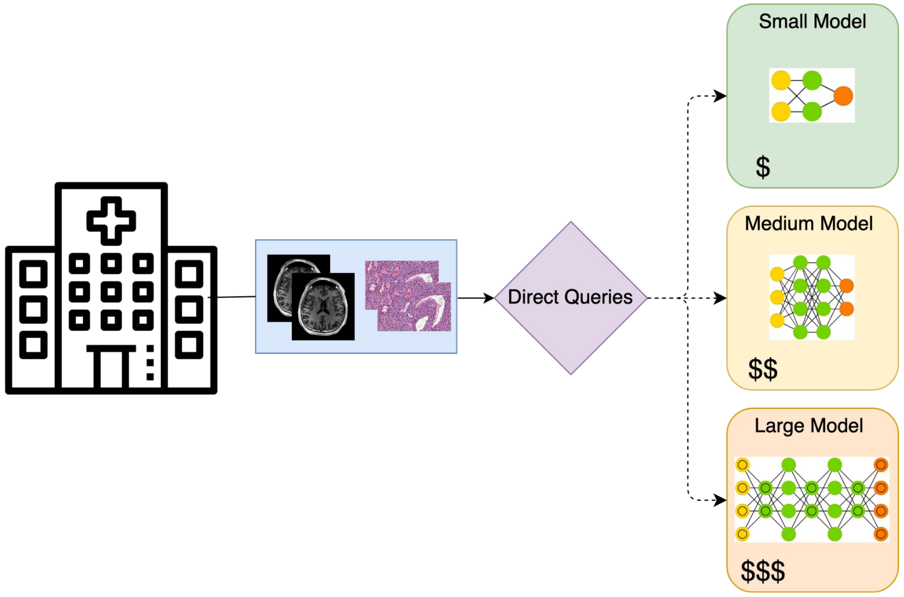
Reinforcement Learning as a Parsimonious Alternative to Prediction Cascades

Bharat Srikishan, Anika Tabassum, Srikanth Allu, Ramakrishnan Kannan, Nikhil Muralidhar

Problem and Motivation

- Deep learning (DL) based models are effective but often come with **increased computational cost** and **high memory requirements**.
- DL combined with Internet of Things (IoT), healthcare, and smart manufacturing means large models are infeasible due to their high computation requirements.
- Previous work has proposed decision cascades (Wang et al. 2017).
- But **cascaded architectures lead to wasted intermediate computation**.
- A more flexible and cost-aware approach can efficiently balance cost with performance.



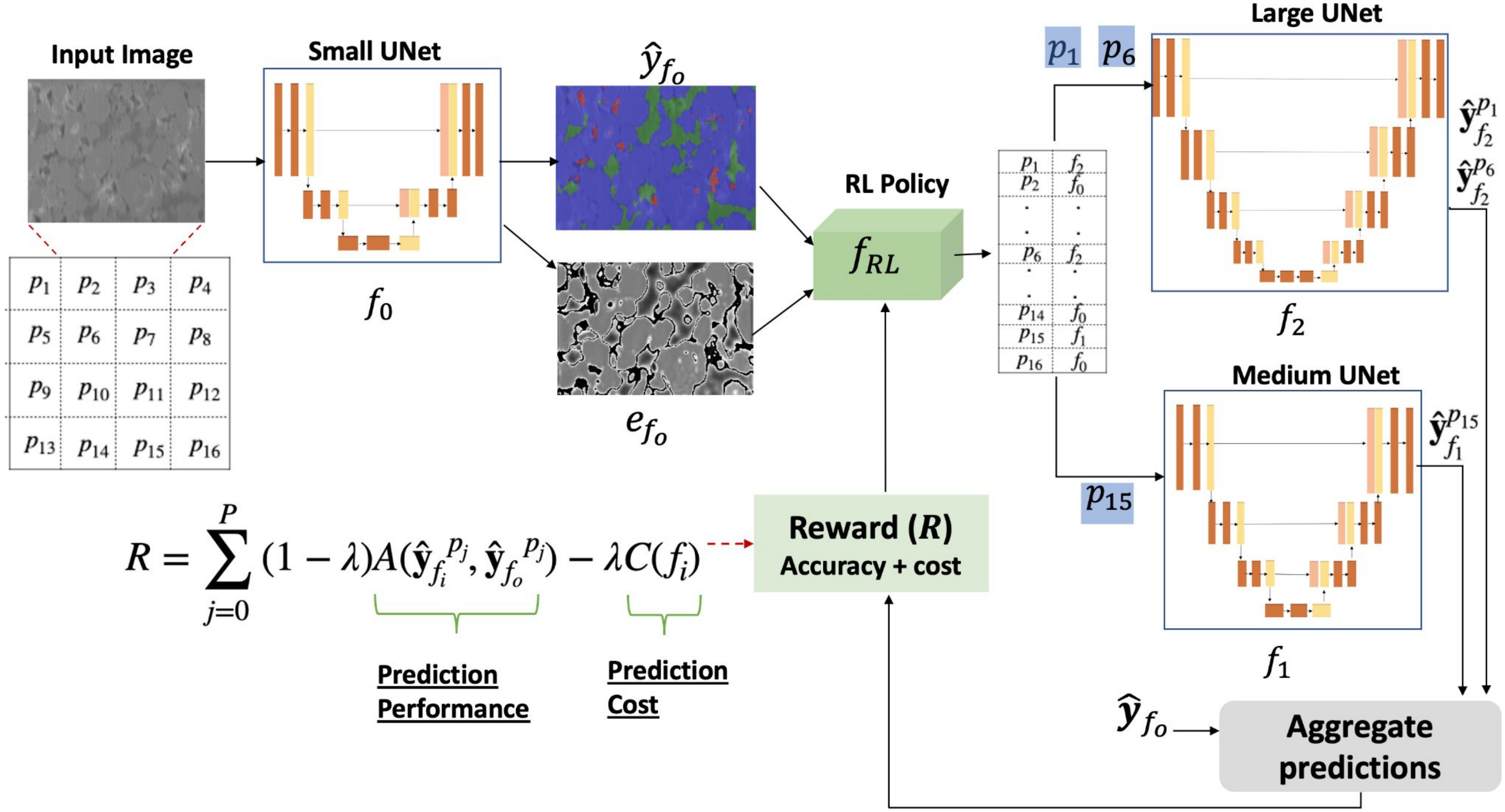


Application: Battery Manufacturing

- Lithium-ion batteries are used in many applications (smartphones, cars, etc)
- The electrode coating of these batteries consist of different material types
- Manufacturing imperfections cause pores to form
- Identifying the pores and materials can determine the quality of the battery
- Researchers have created DL models like MatPhase (Tabassum et al. 2022) to identify materials in battery CT images
- These models are large and expensive to run

PaSeR: Parsimonious Segmentation with RL

- Flexible, cost-aware reinforcement learning (RL) based pipeline as an alternative to a cascaded architecture



PaSeR Reward Function

- Assuming m+1 task models

$$\{f_0, f_1, \ldots, f_m\}$$

- PaSeR optimizes the reward function

$$R(\mathbf{a}) = \sum_{j=0}^{P}(1-\lambda)A(\hat{\mathbf{y}}_{f_{a_j}}^{(p_j)}, \hat{\mathbf{y}}_{f_0}^{(p_j)}) - \lambda C(f_{a_j})$$

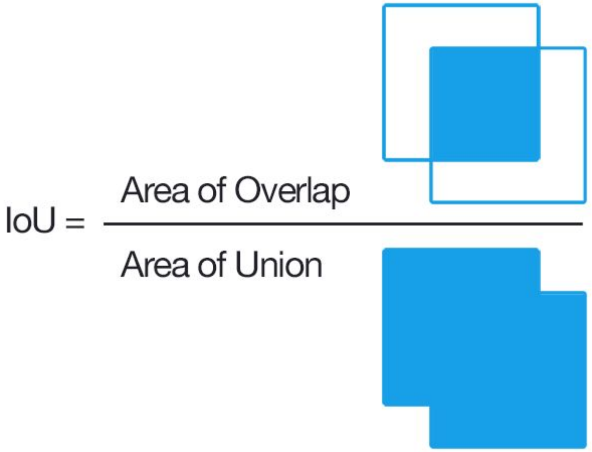
PaSeR Accuracy Function

- For segmentation our accuracy function is:

$$A(\hat{\mathbf{y}}_{f_{a_j}}^{(p)}, \hat{\mathbf{y}}_{f_0}^{(p)}) = IoU(\hat{\mathbf{y}}_{f_{a_j}}^{(p)}, \mathbf{y}^{(p)}) - IoU(\hat{\mathbf{y}}_{f_0}^{(p)}, \mathbf{y}^{(p)})$$

- Intersection over Union (IoU):

$$IoU(A, B) = \frac{|A \cap B|}{|A \cup B|}$$



PaSeR Cost Function

- Our cost function is:

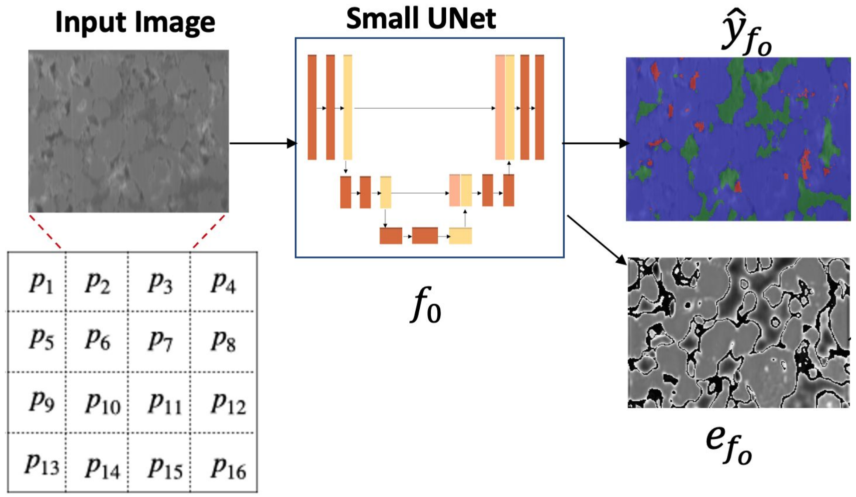
$$C(f_i) = \frac{\text{numParams}(f_i)}{\sum_{j=1}^{m} \text{numParams}(f_j)}$$

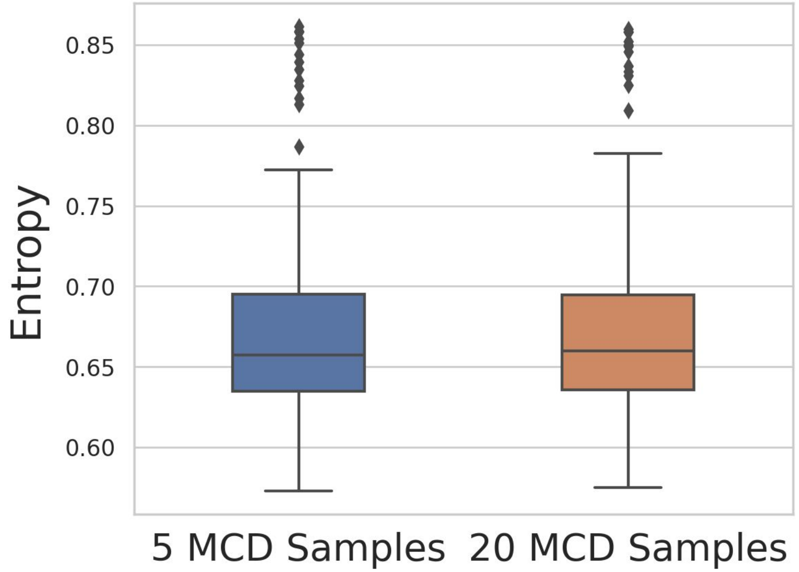
- Use λ to trade-off performance and computation.

$$R(\mathbf{a}) = \sum_{j=0}^{P} (1-\lambda) A(\hat{\mathbf{y}}_{f_{a_j}}^{(p_j)}, \hat{\mathbf{y}}_{f_0}^{(p_j)}) - \lambda C(f_{a_j})$$

Monte-Carlo Dropout Entropy Estimation

- Use Monte Carlo Dropout for entropy estimation
- PaSeR works well with even a small number of samples

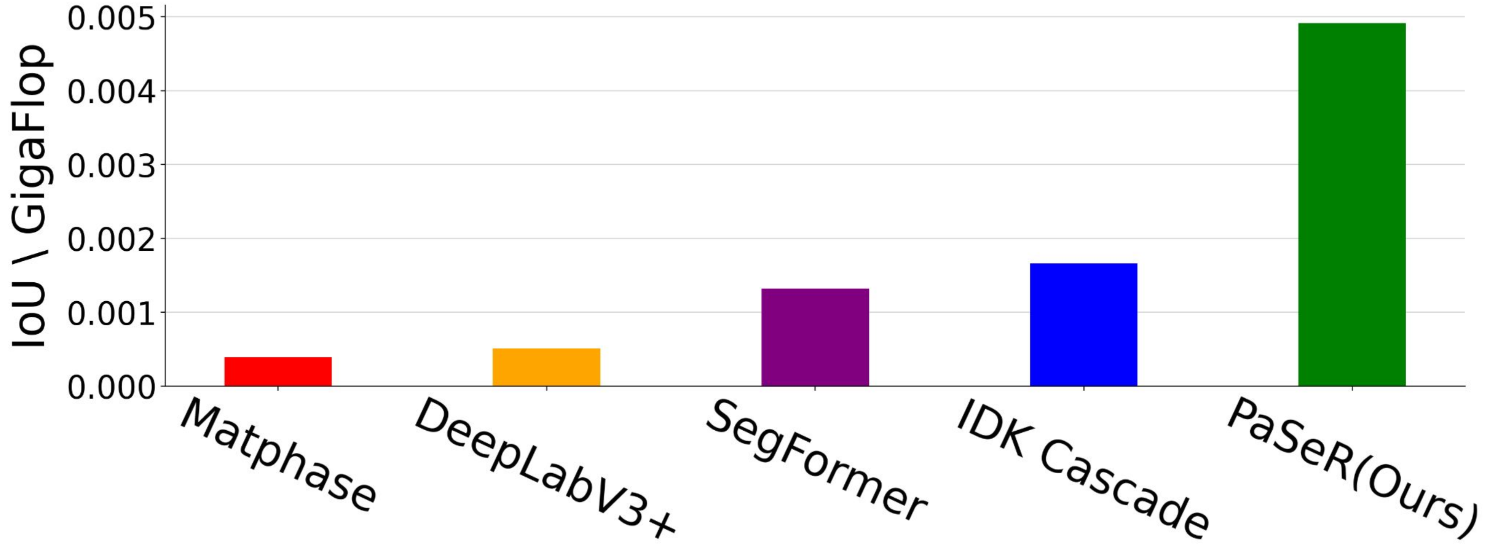




Battery Phase Segmentation Results

- We introduce a novel metric called **IoU/GigaFlop**: the ratio of segmentation performance to computational cost in GigaFlops.

Higher is better



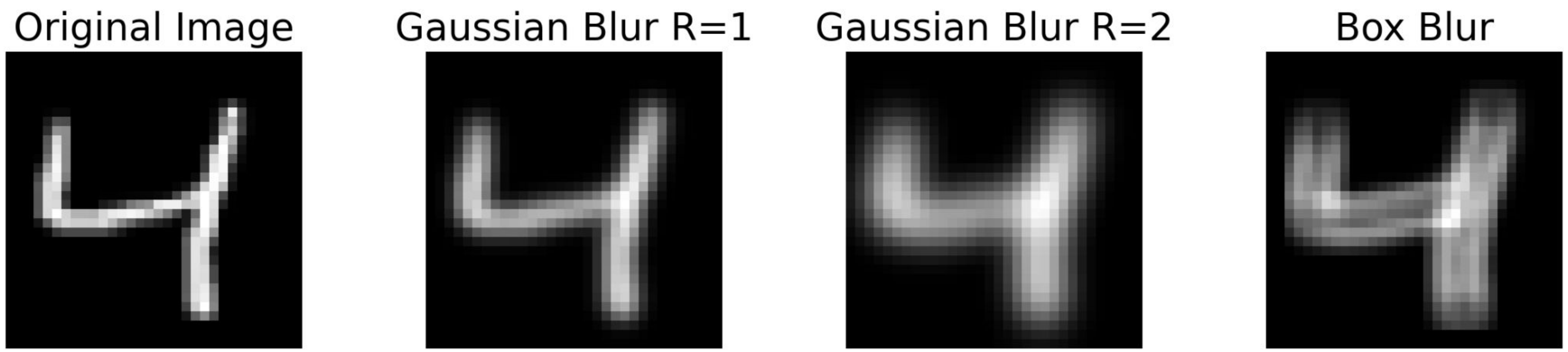
Battery Phase Segmentation Results

- PaSeR outperforms all baselines on the IoU/GigaFlop metric by a minimum of 174% and is within 8% of the best model in terms of IoU.

Model	IoU	Battery Flops	IoU/GigaFlop
Matphase (Tabassum et al. 2022)	0.8144	2.11×10^{12}	0.39×10^{-3}
DeepLabV3+ (Chen et al. 2018b)	0.7817	1.55×10^{12}	0.51×10^{-3}
SegFormer (Xie et al. 2021)	0.7692	5.84×10^{11}	1.32×10^{-3}
EfficientViT (Cai et al. 2022)	0.7765	4.34×10^{11}	1.79×10^{-3}
IDK-Cascade (Wang et al. 2017)	0.6987	4.20×10^{11}	1.66×10^{-3}
PaSeR-RandPol.	0.7234	5.33×10^{11}	1.36×10^{-3}
PaSeR (ours)	0.7426	1.51×10^{11}	$\mathbf{4.91 \times 10^{-3}}$

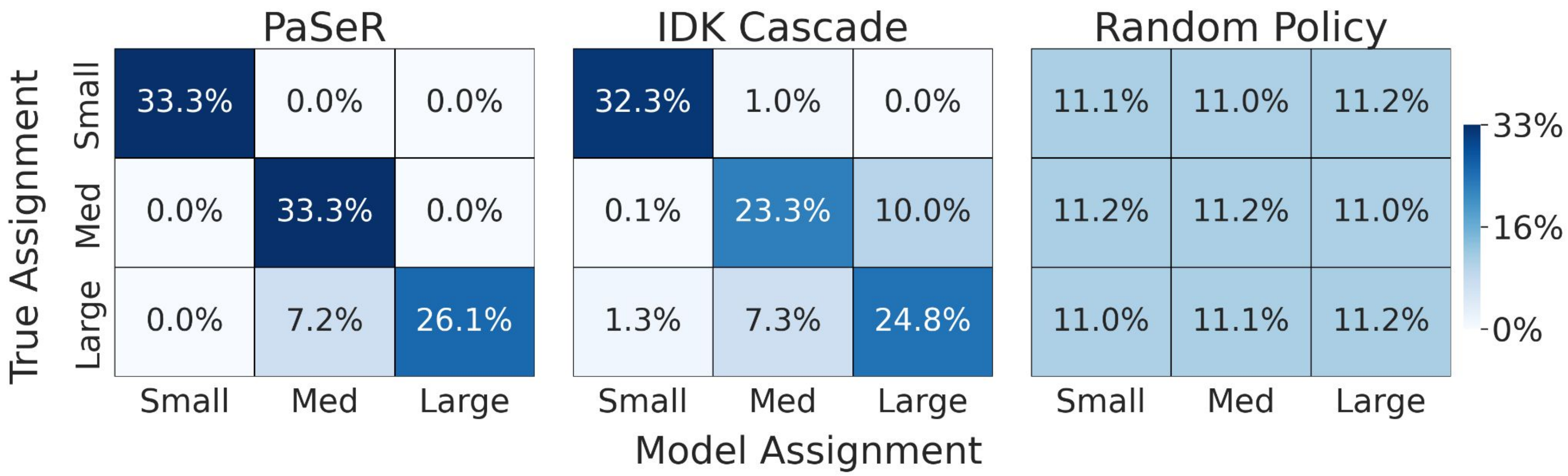
Adaptability to Complementary Models - MNIST

- PaSeR adapts to complementary models
- Add noise to MNIST dataset
- Train each model on its own noise type



Adaptability - MNIST

- PaSeR has nearly perfect assignment
- IDK Cascade model makes more mistakes



Noisy MNIST IoU/GigaFlop Results

- PaSeR outperforms all baselines in terms of IoU/GigaFlop and achieves an IoU that is within 2.3% of the best performing model.

Model	Noisy MNIST		
	IoU	Flops	IoU/GigaFlop
Matphase (Tabassum et al. 2022)	—	—	—
DeepLabV3+ (Chen et al. 2018b)	0.8459	2.07×10^{13}	4.08×10^{-5}
SegFormer (Xie et al. 2021)	0.8448	7.56×10^{12}	1.12×10^{-4}
EfficientViT (Cai et al. 2022)	0.8344	3.72×10^{14}	2.24×10^{-6}
IDK-Cascade (Wang et al. 2017)	0.7750	1.15×10^{13}	6.73×10^{-5}
PaSeR-RandPol.	0.6376	7.05×10^{12}	9.05×10^{-5}
PaSeR (ours)	0.8231	6.51×10^{12}	$\mathbf{1.27 \times 10^{-4}}$

Conclusion

- We develop a novel, **computationally parsimonious RL based model to balance computational cost with task performance**.
- Experiments on **battery phase segmentation data and noisy MNIST data show that PaSeR yields competitive performance with SOTA segmentation models** while also having the highest IoU/GigaFlop.
- We demonstrate the **flexibility of the PaSeR RL policy to adapt to task models with complementary strengths**.
- We introduce a **novel metric IoU/GigaFlop** which measures the segmentation performance obtained per GigaFlop of computation expended.
- Our code is located at: https://github.com/scailab/paser

References

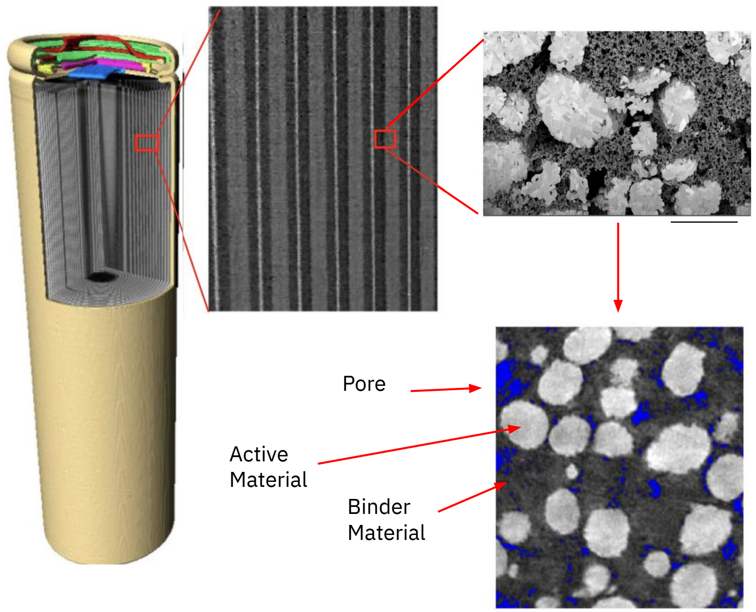
1. Wang, Xin, et al. "Idk cascades: Fast deep learning by learning not to overthink." arXiv preprint arXiv:1706.00885 (2017).
2. Tabassum, Anika, et al. "MatPhase: Material phase prediction for Li-ion Battery Reconstruction using Hierarchical Curriculum Learning." 2022 IEEE International Conference on Big Data (Big Data). IEEE, 2022.
3. Chen, Liang-Chieh, et al. "Encoder-decoder with atrous separable convolution for semantic image segmentation." Proceedings of the European conference on computer vision (ECCV). 2018.
4. Xie, Enze, et al. "SegFormer: Simple and efficient design for semantic segmentation with transformers." Advances in Neural Information Processing Systems 34 (2021): 12077-12090.
5. Cai, Han, et al. "EfficientViT: Multi-Scale Linear Attention for High-Resolution Dense Prediction." arXiv preprint arXiv:2205.14756 (2022).
6. Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." International Conference on Machine Learning. PMLR, 2016.

Thank You!

Application: Battery Manufacturing

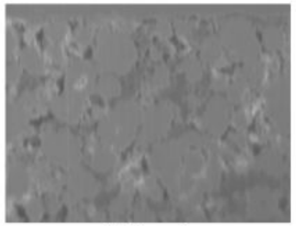
- Lithium-ion batteries are used in many industrial applications.
- The electrode coating of these battery cathode consists of active materials and polymeric binders.
- Due to the imperfections during manufacturing, small pores are also present.
- Finding phase transitions of these active materials, binders, and pores helps to estimate the overall quality of the battery.
- Researchers have created DL models like MatPhase (Tabassum et al. 2022) which use low resolution CT images of the battery to identify the composite battery materials.
- This model is computationally expensive to run at inference time.

X-ray Nano-CT Image



Full Reconstruction is **Expensive** but Accurate

*Image Courtesy(Lu et al. 2020)



Cheaper X-ray Micro-Tomography Image poses for harder / **significantly less accurate reconstruction**

Reinforcement Learning as a Parsimonious Alternative to Prediction Cascades

- Recent advances in Deep Learning (DL) have lead to state of the art (SOTA) performance on computer vision tasks such as object detection, classification, and image segmentation.
- Many of these modern DL models are large (over-parameterized) and monolithic.
- While these large models lead to better performance, they are computationally expensive and memory intensive even during inference.
- In settings such as smart manufacturing, we cannot afford to always run such large models on the factory floor.
- For many modern applications, we have a hierarchy of compute where efficient small models are locally available while large models are in the cloud.