Adaptive Multi-Exit Neural Networks using EM-Based Routing

Jason Thwe Kyauk[1]

Department of Computer Science, Stanford University

Stanford
Computer Science

Project Overview

- Real-time vision systems treat all inputs as computationally equivalent, despite some requiring less compute and undergoing layers for accurate predictions.
- Most Adaptive Compute methods are heuristic-based [1],[2],[5]
- Work has been done with a probabilistic lens for halting decisions [3],[4]
- This project proposes a probabilistic routing strategy instead, reframing early-exiting as an inference problem
- Routers are trained on posterior distributions of the data to predict which exit to take

Datasets & Metrics

- CIFAR-10 Dataset
    - Contains 60k 32x32 images of 10 distinct objects
    - Project success was determined by accuracy of classification, and expected compute of the multi-exit network.

References

[1] Teerapittayanon, S., McDanel, B., & Kung, H. (2017). BranchyNet: Fast inference via early exiting. arXiv:1709.01686.

[2] Huang, G., Chen, D., Li, T., Wu, F., van der Maaten, L., & Weinberger, K. Q. (2018). Multi-scale Dense Networks for Resource Efficient Image Classification. arXiv:1703.09844.

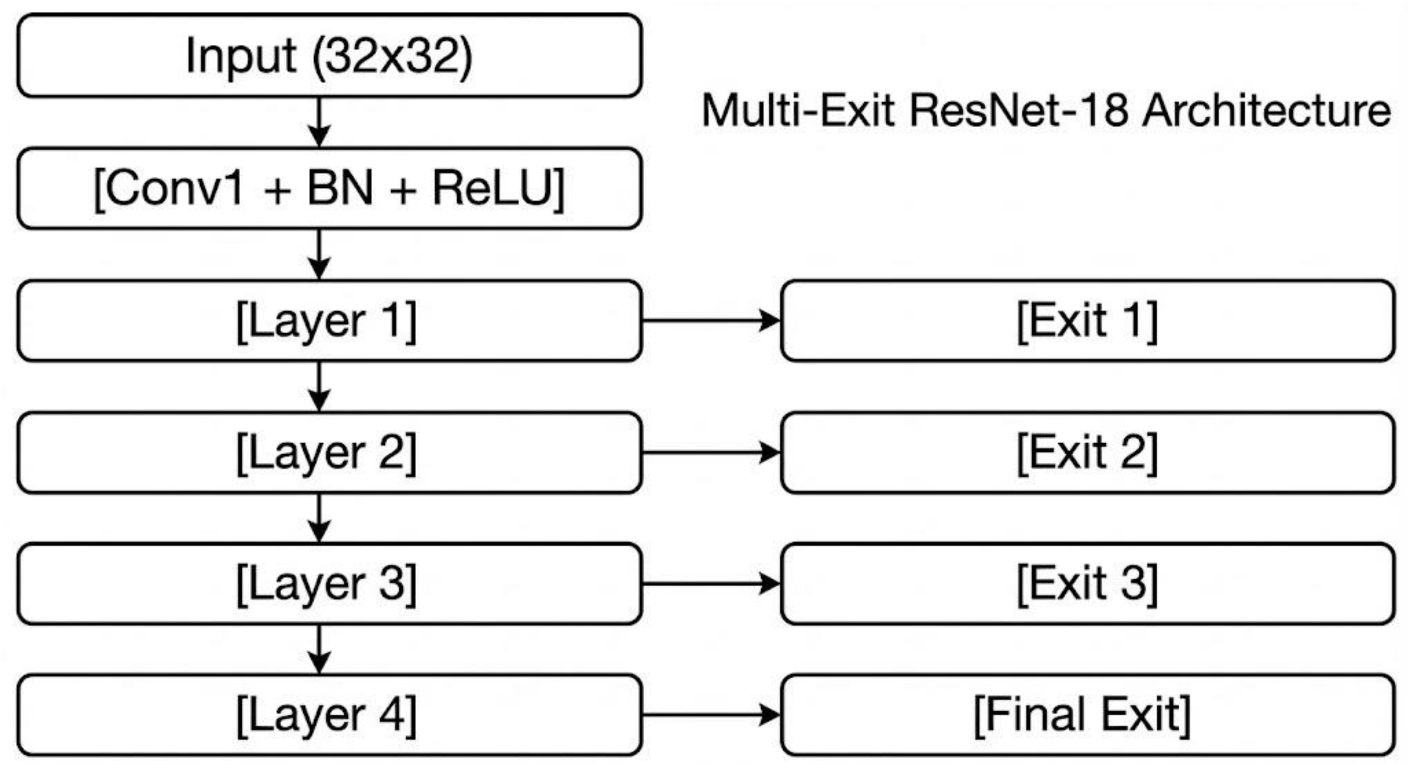
[3]Graves, A. (2016). Adaptive Computation Time for Recurrent Neural Networks. arXiv:1603.08983.

[4]Sukhbaatar, S., Xu, Z., Vinyals, O., & Denoyer, L. (2023). Adaptive Computation with Elastic Input Sequence arXiv:2301.13195.

[5] E. Demir and E. Akbas, "Early-exit Convolutional Neural Networks," *arXiv preprint arXiv:2409.05336*, 2024.

Methods & Experiments

- Used frozen ResNet-18 backbone that's been retrained on CIFAR-10 and added 4 exit layers, trained via CE-Loss
- Initialized probability of each exit layer being chosen as 0.25 (1 / # of exits)
- Used EM-Algorithm to uncover posterior distributions for each exit per training example
- Trained routers (1 per exit) to predict posterior distributions at given exit
- Tested against a Regular ResNet-18, Randomized Exits, Fixed Exits, Confidence-Threshold based model, and an oracle model



Multi-Exit ResNet-18 Architecture

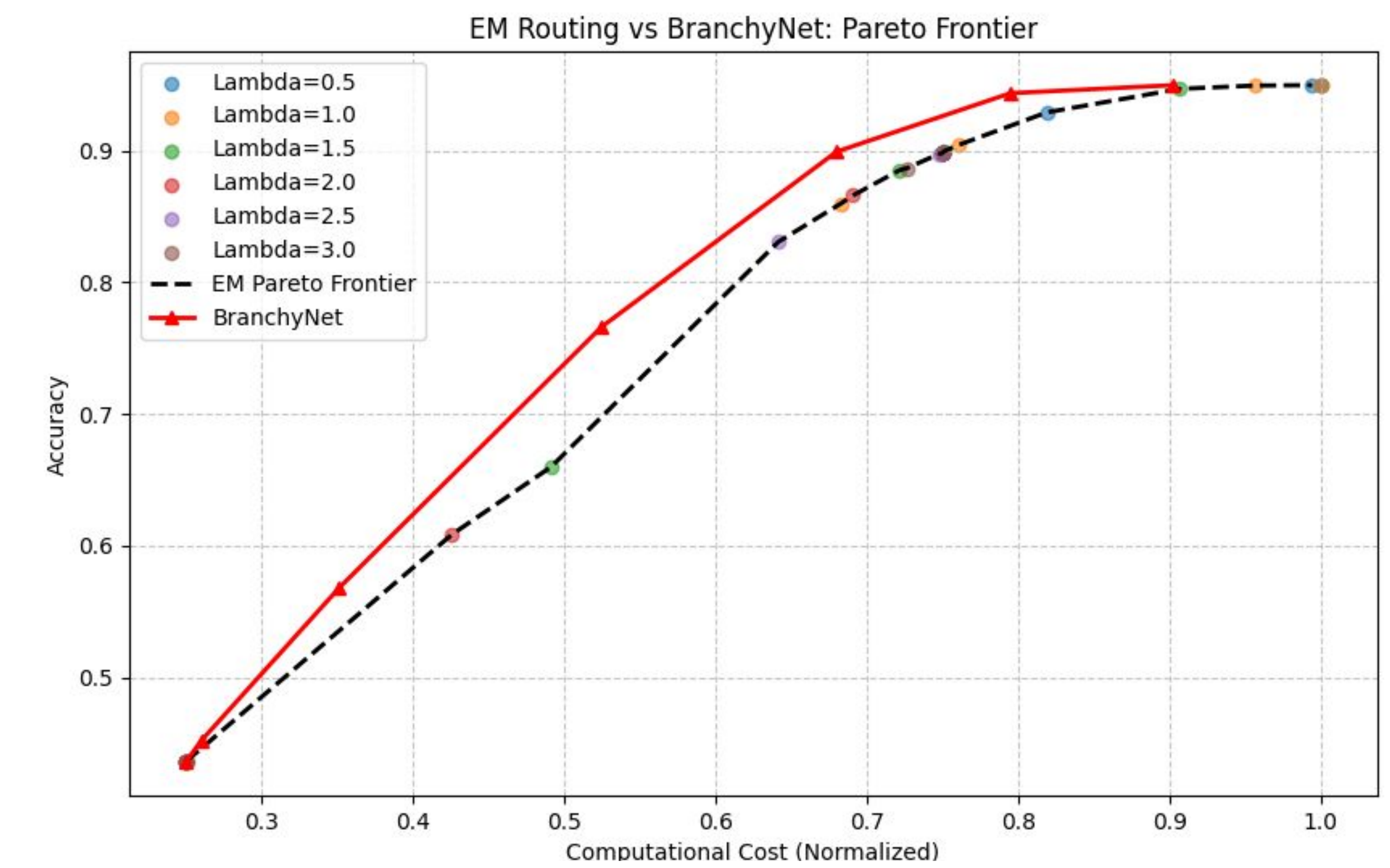
Discussions & Future Research

- EM-Routing method worked well because posterior distributions aligned with example difficulty
- Eliminated confined view of a single exit layer, and allowed model a more holistic view
- Fell short however since exits were not jointly-optimized, limiting model's context
- Router networks also predicted exit assignments rather labels, therefore we ultimately had to set thresholds eventually, but rather posterior distribution-thresholds.
- Future work points towards finding a way to jointly optimize the network, and eliminating threshold in its entirety to maximize the information gained from the posterior distribution
- Future work also includes testing on other expansive datasets beyond CIFAR-10

Results

Method	Accuracy	Cost
ResNet-18	0.9497	1.0000
Exit 1 Only	0.4359	0.2500
Exit 2 Only	0.6635	0.5000
Exit 3 Only	0.8984	0.7500
Exit 4 Only	0.9497	1.0000
Random Routing	0.7501	0.6384
BranchyNet	**0.8756**	**0.6452**
EM Routing	**~0.8384**	**0.65**
Oracle Routing	0.9691	0.4855

EM-Routing vs Confidence-Based Thresholding



Lambda affects Accuracy-Cost