Mechanizmy dynamicznej adaptacji sieci neuronowych

Dynamic neural networks

Mikołaj Małkiński
mikolaj.malkinski.dokt@pw.edu.pl

April 12, 2023

Warsaw University of Technology
Faculty of Mathematics and Information Science

Table of contents

Han, Yizeng, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang. **"Dynamic neural networks: A survey."** IEEE Transactions on Pattern Analysis and Machine Intelligence 44, no. 11 (2021): 7436-7456. [5]

Motivation

1. Typical neural networks have static computation graph and parameters after training.
2. Dynamic neural networks can adapt their structure and parameters during inference.

Dynamic neural networks

1. **Efficiency:** Can allocate computation during inference conditioned on the input.
2. **Representation power:** Data-dependent structure or parameters enlarge the parameter space.
3. **Adaptiveness:** Allow to achieve a trade-off between accuracy and efficiency.

Dynamic neural networks

4. **Compatibility:** Dynamic mechanisms are often orthogonal to advancements in other methods.
5. **Generality:** Mechanisms of adaptation can be transferred between problem domains.
6. **Interpretability:** Adaptation offers another axis for interpreting the models.

Sample-wise models

Sample-wise models

Based on each sample:

1. Adjust architecture to appropriately allocate computation.
2. Adapt parameters to increase representational power.

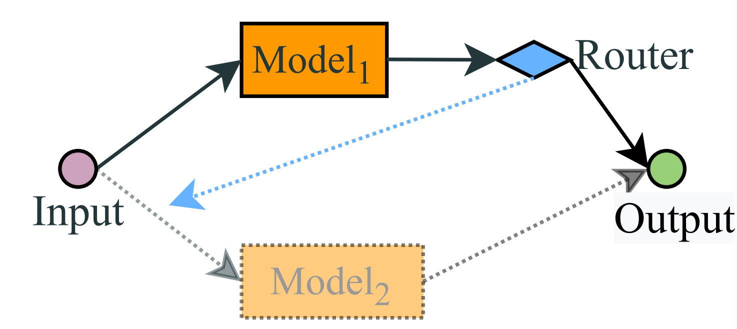

Figure 1: Cascading of models.

Early exiting – model cascade

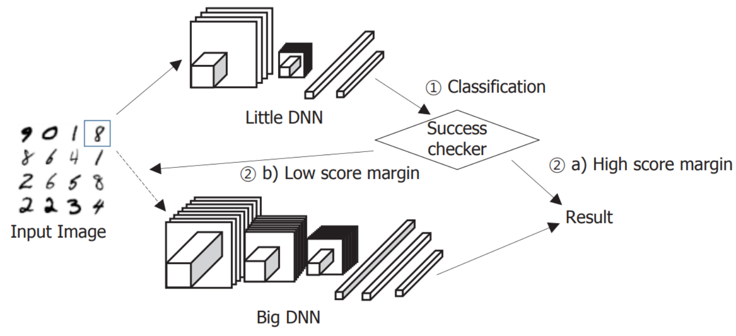

Figure 2: Big/Little Deep Neural Network [11].

Early exiting – model cascade

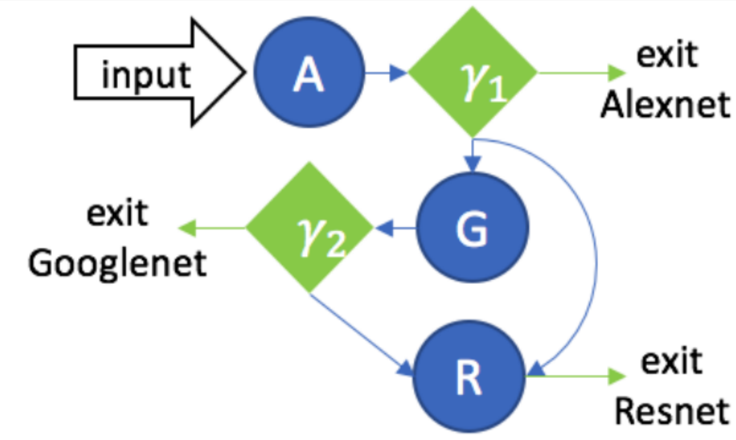

Figure 3: Network topology selection for AlexNet, GoogleNet, and ResNet [1].

Early exiting – intermediate classifiers

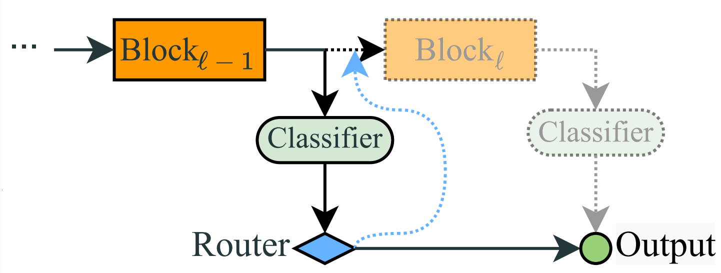

Figure 4: Network with intermediate classifiers.

Early exiting – intermediate classifiers

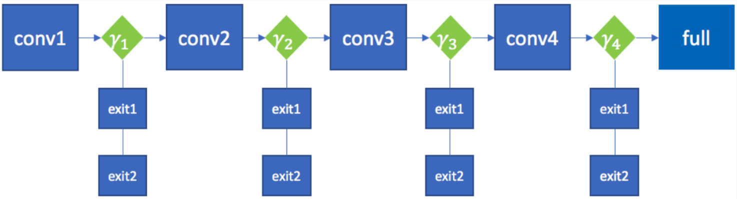

Figure 5: Early exiting system for AlexNet [1].

Early exiting – intermediate classifiers

Drawbacks:

1. Classifiers can interfere with each other.
2. High-resolution features lack high-level information required for classification.
3. Early classifiers can force the shallow layers to produce task-specific features.

Early exiting – multi-scale processing

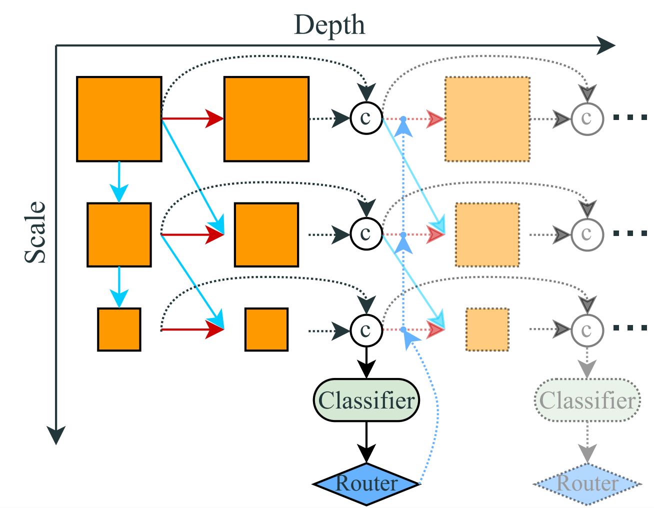

Figure 6: Multi-scale Dense Network (MSDNet) [6].

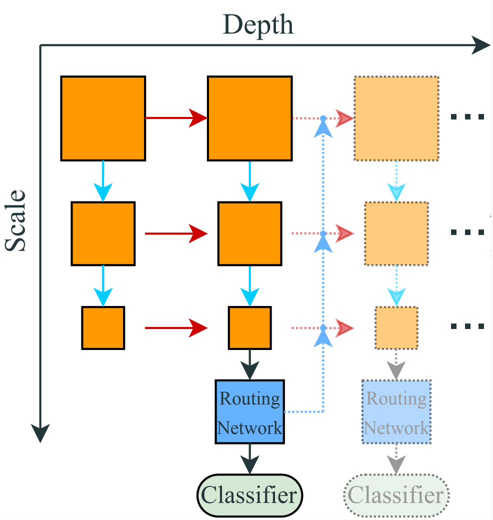

Figure 7: Routing networks [10].

Early exiting – multi-scale processing

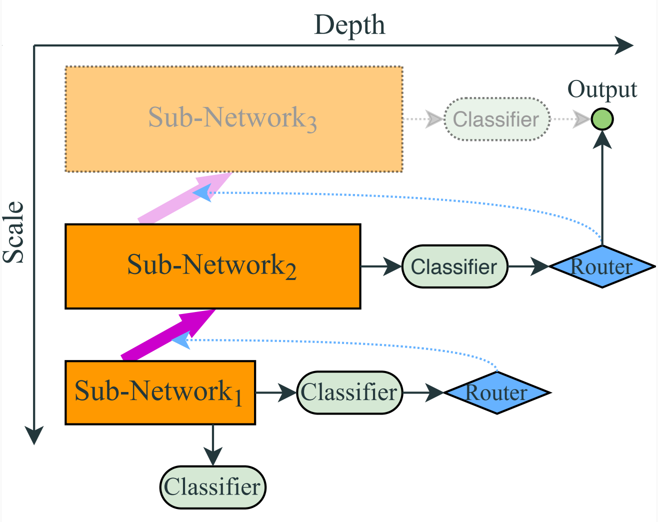

Figure 8: Resolution Adaptive Network (RANet) [15].

Early exiting – multi-scale processing

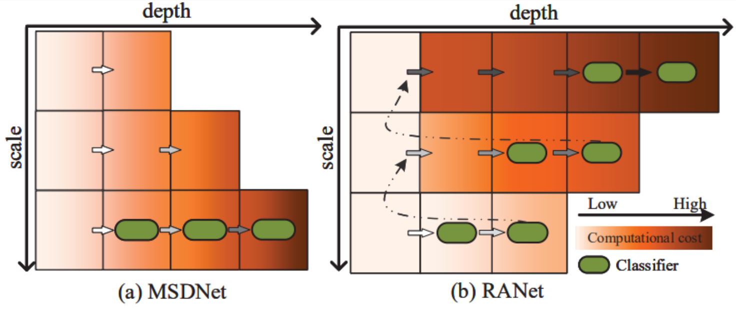

Figure 9: MSDNet vs RANet [15]

Early exiting vs layer skipping

1. **Early exiting:** skip execution of layers after a certain classifier.
2. **Layer skipping:** skip execution of intermediate layers.

Layer skipping – halting score

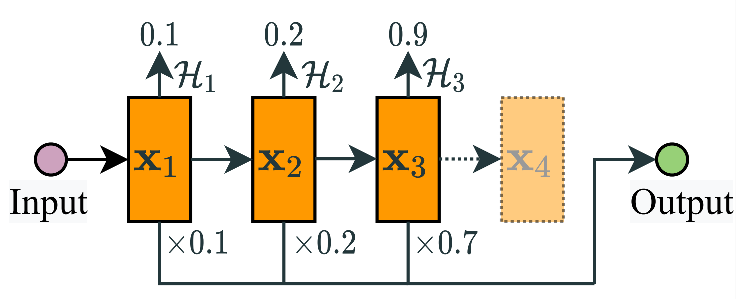

Figure 10: Halting score.

Layer skipping – halting score

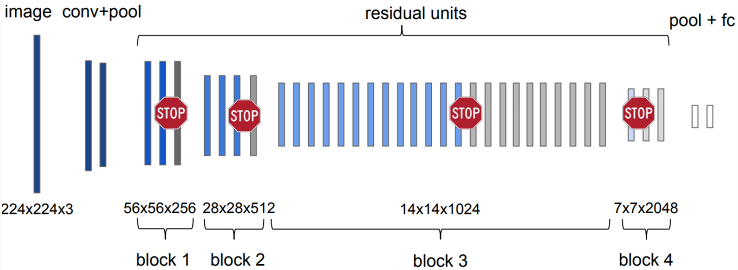

Figure 11: Adaptive skipping of layers in ResNet based on a halting score [4].

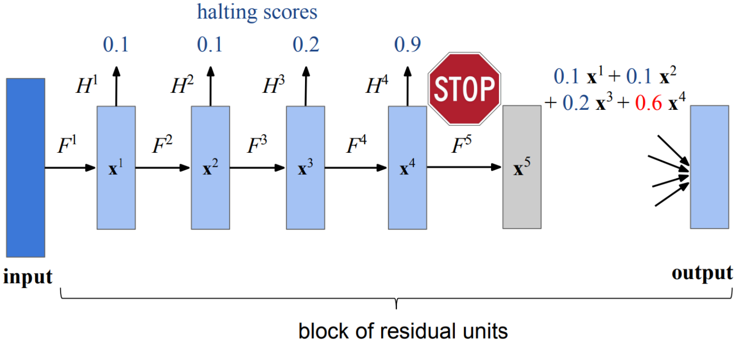

Figure 12: Adaptive computation time for one block of residual units in ResNet [4].

Layer skipping – halting score

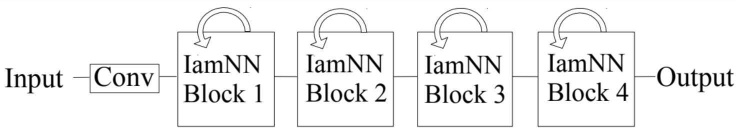

Figure 13: Layers in ResNet can be adaptively repeated based on an adaptive computation time [9].

Layer skipping – gating function

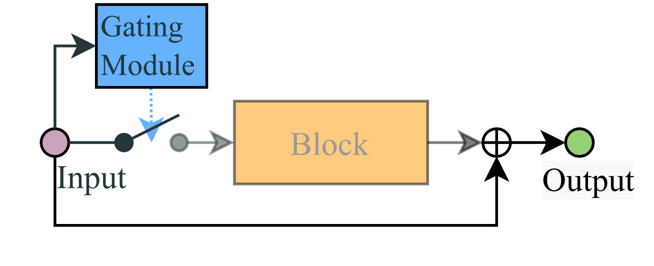

Figure 14: Gating function.

Layer skipping – gating function

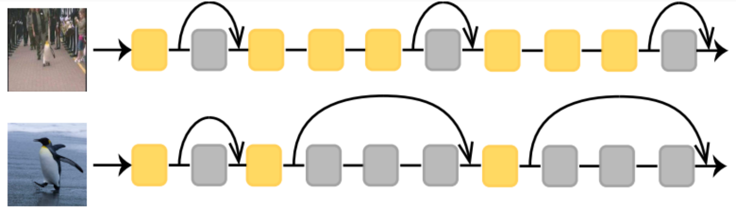

Figure 15: SkipNet learns to skip certain convolutional layers with a gating module [13].

Layer skipping – policy network

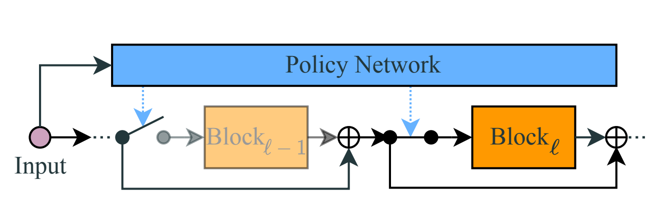

Figure 16: Policy network.

Layer skipping – policy network

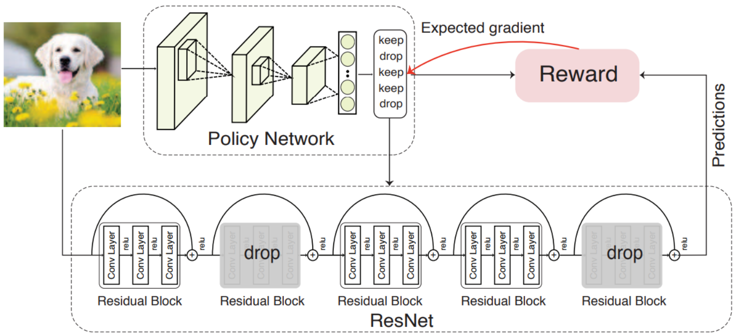

Figure 17: Policy Network predicts drop / keep decisions for the layers of a ResNet [14].

Dynamic depth vs dynamic width

1. **Dynamic depth:** adapt the number of executed layers.
2. **Dynamic width:** adjust the number of units (e.g., neurons, branches) executed in a given layer.

Cascade of models vs mixture of experts

1. **Cascade of models:** models are executed serially.
2. **Mixture of experts (MoE) [7]:** modules are run in parallel and their output is fused.

Mixture of experts – soft weighting

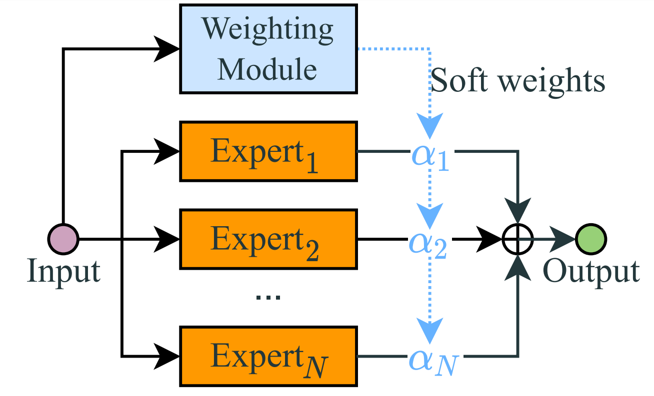

Figure 18: Soft attention in MoE.

Mixture of experts – hard weighting

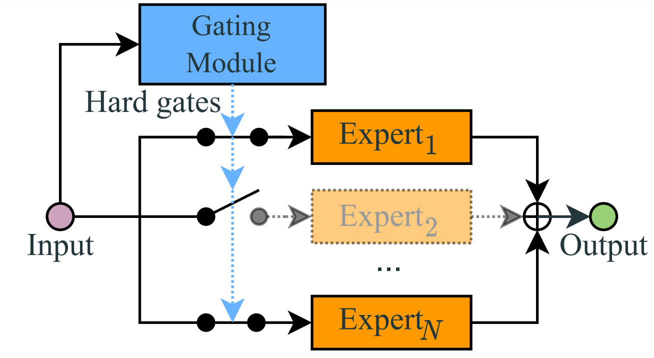

Figure 19: Hard attention in MoE.

Mixture of experts: soft weighting vs hard weighting

1. **Soft weighting:** all experts have to be executed even in test time.
2. **Hard weighting:** computation can be limited to experts with non-zero weights.

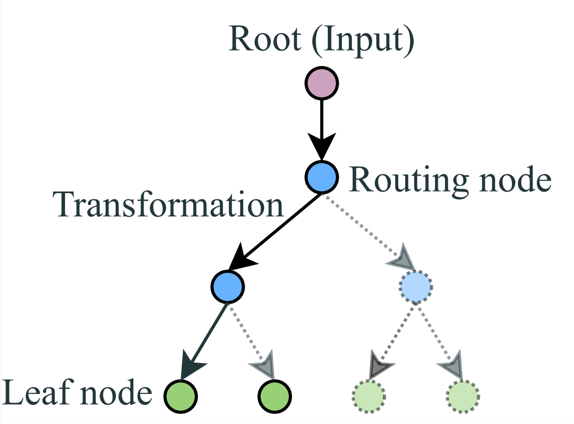

Figure 20: Tree structure.

Dynamic structure vs dynamic parameters

1. **Dynamic structure:** efficient allocation of resources, but might require custom training strategies.
2. **Dynamic parameters:** minor increase in computational cost, but allows to increase representational power.

Dynamic parameters

Main research directions:

1. Adapt trained parameters based on the input.
2. Directly generate parameters based on the input.
3. Rescale features with soft attention.

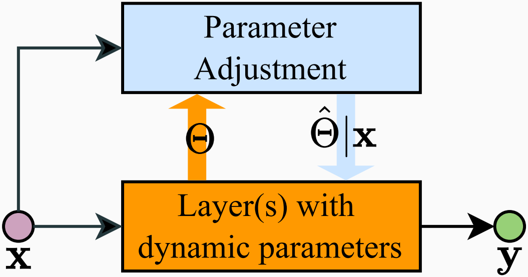

Figure 21: Dynamic weight adjustment.

Dynamic parameters – weight adjustment

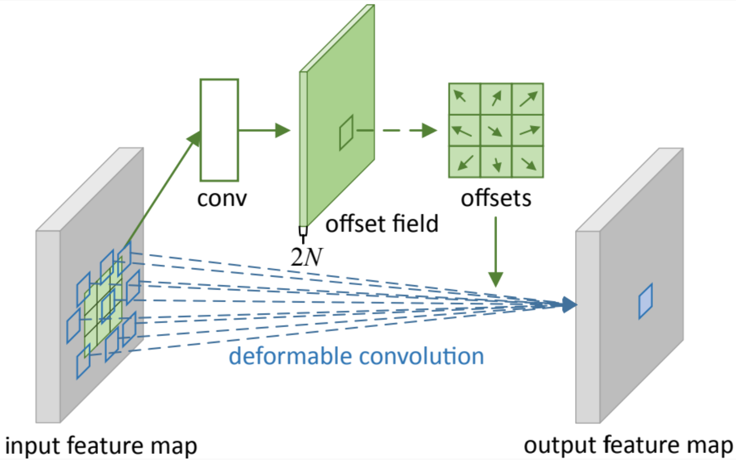

Figure 22: Deformable convolution adjusts offsets for spatial sampling locations of a convolution filter [3].

Dynamic parameters – weight adjustment

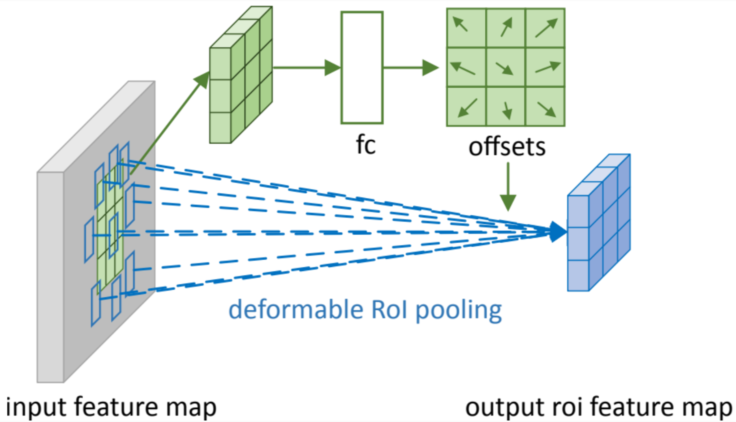

Figure 23: Deformable pooling adjusts offsets for spatial sampling locations of a pooling operation [3].

Dynamic parameters – weight adjustment

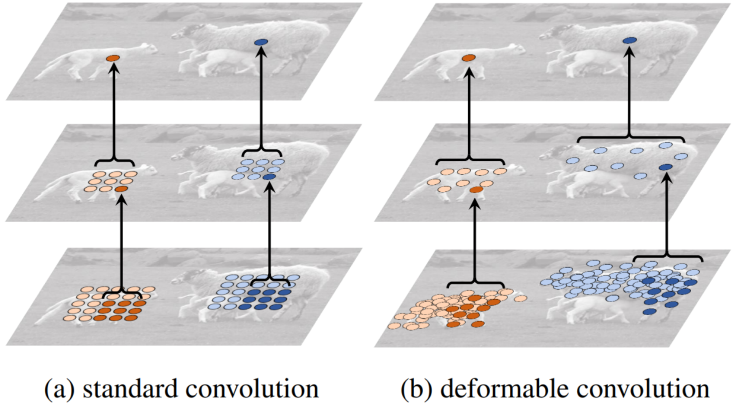

Figure 24: Deformable convolution enables image processing with an adaptive receptive field [3].

Dynamic parameters – weight prediction

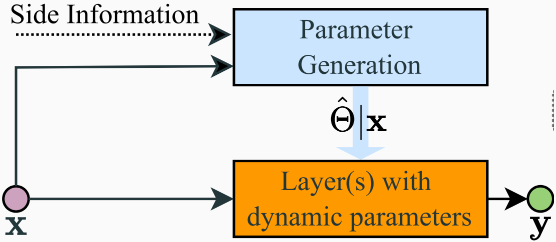

Figure 25: Dynamic weight prediction.

Dynamic parameters – weight prediction

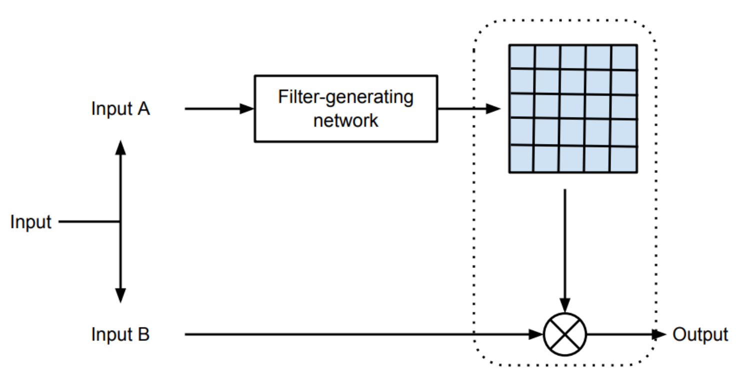

Figure 26: Dynamic Filter Network predicts convolution filters dynamically based on the input [8].

Dynamic parameters – soft attention

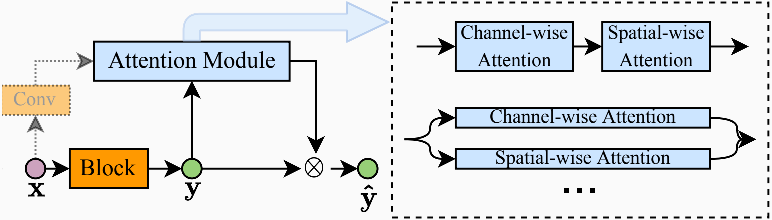

Figure 27: Soft attention.

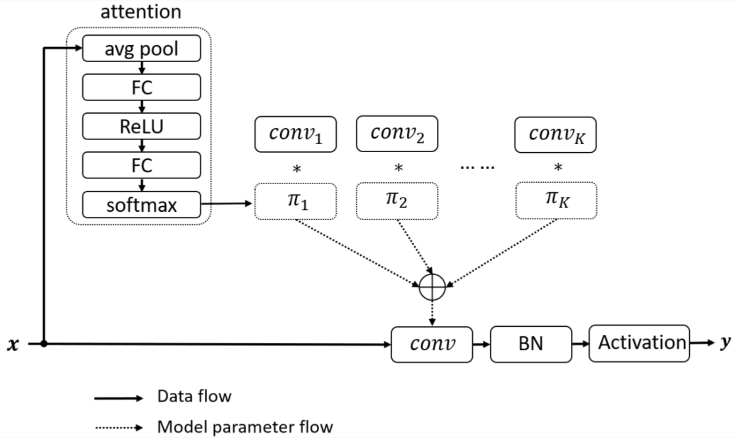

Figure 28: Dynamic Convolution aggregates multiple convolution kernels based on the input [2].

Dynamic parameters

1. **Weight adjustment / prediction:** increase representational power with a small increase in the number of parameters.
2. **Soft attention:** increase model complexity without increasing depth or width.

Spatial-wise models

Spatial-wise models

1. Not all image locations are equally relevant in computer vision.
2. Spatial dynamic computation can eliminate some redundancy.
3. Such models adapt computation differently to spatial locations.

Spatial-wise models – dynamic convolution

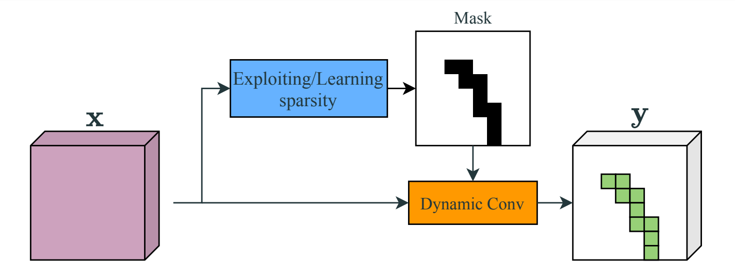

Figure 29: Dynamic convolution on selected spatial locations.

Spatial-wise models – dynamic convolution

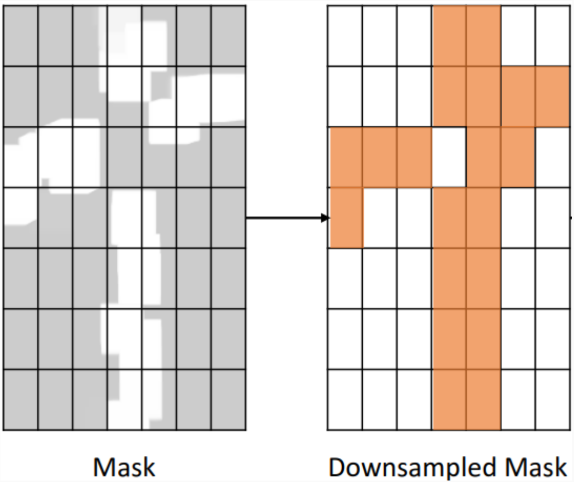

Figure 30: Efficient algorithms for processing sparse matrices are required [12].

Spatial-wise models – region-level dynamic inference

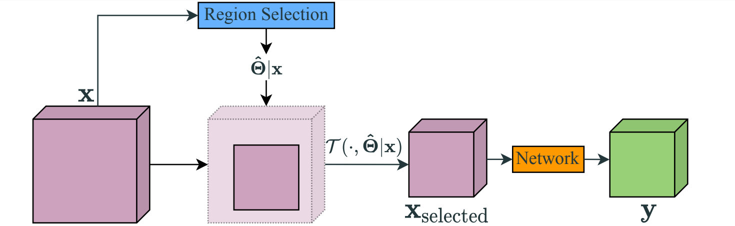

Figure 31: Region-level dynamic inference.

Temporal-wise models

Temporal adaptive inference – skip update

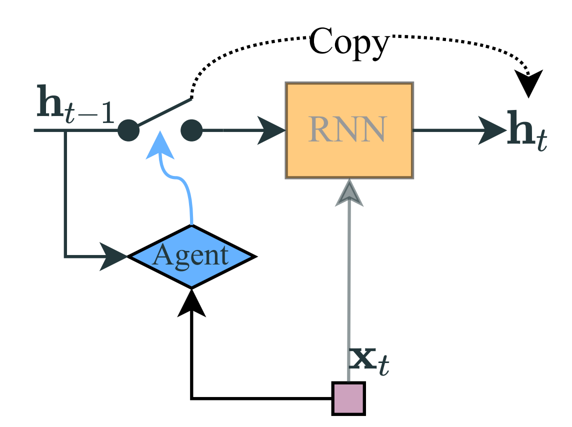

Figure 32: Skip update of a hidden state.

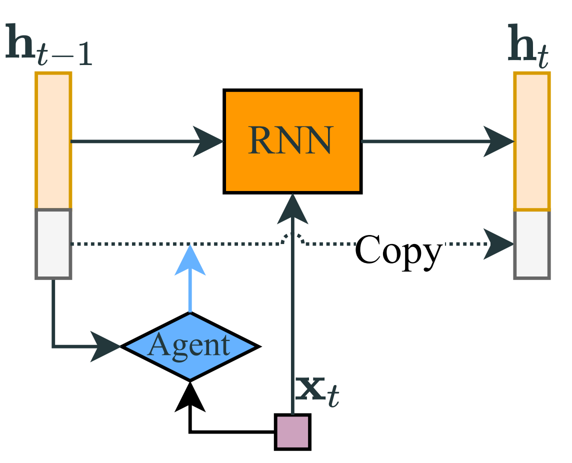

Figure 33: Partial update of a hidden state.

Temporal adaptive inference – skip tokens

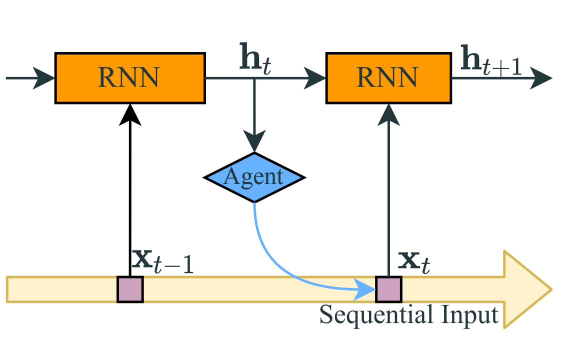

Figure 34: Temporal dynamic jumping.

Temporal adaptive inference – skip tokens

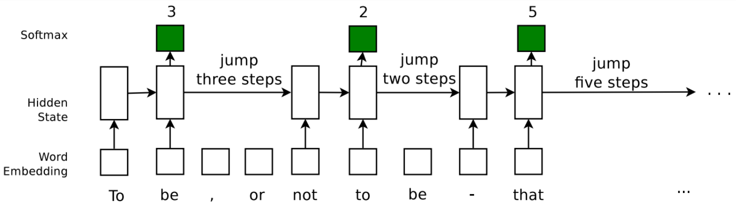

Figure 35: Adaptive mechanism in an RNN decides how many input tokens to skip [16].

Training and inference

Decision making mechanisms – sumary

1. Confidence-based criteria
2. Policy networks
3. Gating functions

Training dynamic networks

1. **Multi-exit:** minimize weighted cumulative loss of all classifiers.
2. **Encourage sparsity:** minimize an auxiliary loss that promotes sparsity.

Q&A

Tolga Bolukbasi, Joseph Wang, Ofer Dekel, and Venkatesh Saligrama.
Adaptive neural networks for efficient inference.
In *International Conference on Machine Learning*, pages 527–536. PMLR, 2017.

Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu.
Dynamic convolution: Attention over convolution kernels.
In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11030–11039, 2020.

Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei.
Deformable convolutional networks.
In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

Michael Figurnov, Maxwell D Collins, Yukun Zhu, Li Zhang, Jonathan Huang, Dmitry Vetrov, and Ruslan Salakhutdinov.
Spatially adaptive computation time for residual networks.
In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1039–1048, 2017.

Yizeng Han, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang.
Dynamic neural networks: A survey.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(11):7436–7456, 2021.

Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, Laurens van der Maaten, and Kilian Weinberger.
Multi-scale dense networks for resource efficient image classification.
In *International Conference on Learning Representations*, 2018.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton.
Adaptive mixtures of local experts.
Neural computation, 3(1):79–87, 1991.

Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool.
Dynamic filter networks.
Advances in neural information processing systems, 29, 2016.

Sam Leroux, Pavlo Molchanov, Pieter Simoens, Bart Dhoedt, Thomas Breuel, and Jan Kautz.
Iamnn: Iterative and adaptive mobile neural network for efficient image classification.
arXiv preprint arXiv:1804.10123, 2018.

Mason McGill and Pietro Perona.
Deciding how to decide: Dynamic routing in artificial neural networks.
In *International Conference on Machine Learning*, pages 2363–2372. PMLR, 2017.

Eunhyeok Park, Dongyoung Kim, Soobeom Kim, Yong-Deok Kim, Gunhee Kim, Sungroh Yoon, and Sungjoo Yoo.
Big/little deep neural network for ultra low power inference.
In *2015 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ ISSS)*, pages 124–132. IEEE, 2015.

Mengye Ren, Andrei Pokrovsky, Bin Yang, and Raquel Urtasun.
Sbnet: Sparse blocks network for fast inference.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8711–8720, 2018.

Xin Wang, Fisher Yu, Zi-Yi Dou, Trevor Darrell, and Joseph E Gonzalez.
Skipnet: Learning dynamic routing in convolutional networks.
In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 409–424, 2018.

Zuxuan Wu, Tushar Nagarajan, Abhishek Kumar, Steven Rennie, Larry S Davis, Kristen Grauman, and Rogerio Feris.
Blockdrop: Dynamic inference paths in residual networks.
In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8817–8826, 2018.

Le Yang, Yizeng Han, Xi Chen, Shiji Song, Jifeng Dai, and Gao Huang.
Resolution adaptive networks for efficient inference.
In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2369–2378, 2020.

Adams Wei Yu, Hongrae Lee, and Quoc V Le.
Learning to skim text.
arXiv preprint arXiv:1704.06877, 2017.