Novel Neural Architectures & Algorithms for Efficient Inference

Researchers have trained large Deep Neural Networks (DNNs) due to advances in training algorithms, hardware infrastructure, and neural constructs. For instance, convolutional models [25] for image classification, transformers [13] for natural language processing, diffusion models [23] for mixed modalities, etc. Although these models achieve state-of-the-art (SOTA) performance, they have enormous storage, computational, and energy requirements. This paradigm of throwing vast resources for training larger and larger models has the following negative impacts:

- **Slow Inference on Commodity Hardware.** Without access to GPU/TPU clusters, it would be impossible to perform fast inference with these models, let alone training.
- **Prohibitively Large Model Size.** High-capacity DNNs have large model storage requirements. Thus, they cannot be deployed on resource-constrained devices such as mobile phones.
- **High Carbon Footprint.** Soaring energy consumption yields high CO_2 emissions [14].

I am broadly interested in improving the resource efficiency of neural architectures, i.e., significantly reducing the computational and storage requirements of a DNN without any loss in performance. I have proposed novel architectures and algorithms that allow low-capacity models to achieve near SOTA performance. Specifically, my research focuses on the following two themes:

- **Efficient Low-Complexity Architectures.** In this theme, the fundamental problem is achieving better performance given a resource constraint such as storage or computation. I tackled this issue in two ways. First, I have designed better-performing recurrent and convolutional architectures than existing networks using differential equations. Second, I have proposed training strategies that promote better performance and resource usage trade-off efficiency, even in the existing architectures.
- **Input Hardness Aware Architectures.** In the previous theme, a DNN spends its full computational power for every input without regard to input characteristics. In contrast, a model aware of the input hardness would spend considerably fewer resources on easy inputs than difficult ones. I have built a series of architectures that incorporate this notion. First, I have developed an abstaining classifier that skips prediction on a few uncertain examples. Next, I have trained hybrid models wherein a low-capacity abstaining network sends the abstained inputs to a high-capacity network. Finally, I designed a distillation training method wherein a helper function selectively distills teacher knowledge onto the student only on easy inputs.

Below, I will describe these two research directions, highlighting my previous work. Finally, I will lay out my future research agenda.

1 Efficient Low Complexity Architectures

Existing DNNs, such as Recurrent Neural Networks (RNNs) [17] and Convolutional Neural Networks (CNNs) [16], suffer from many issues like poor gradients, low receptive fields, low signal-to-noise ratio, etc. These instabilities lead to sub-optimal performance. A simple strategy to achieve better performance at a given resource constraint is to address these instabilities, leading to improved architectures and training algorithms. Below, I summarize these challenges posed by RNNs and CNNs and describe my previous work to remedy the same.

- **RNNs : Vanishing/Exploding Gradients.** During RNN training, the gradient of loss backpropagated in time could suffer from exponential decay/explosion [21], resulting in poor generalization for processes exhibiting long-term dependencies. I have designed Incremental RNN [10], a continuous time RNN whose transition function is a solution to an ordinary differential equation (ODE). It tracks the increments in the hidden state space. As a result, the ODE equilibrium yields identity gradients and eliminates the vanishing and exploding gradients. Further, a simple few-step Euler discretization efficiently processes long sequences while achieving significantly better performance than existing architectures with 4× storage and 2× compute reduction.

- **RNNs : Low Signal-to-Noise Ratio.** Sequential data such as time-series has a low signal-to-noise ratio. Thus, improperly designed RNNs such as LSTMs [17] could amplify noise in the hidden states. I have further improved Incremental RNNs by developing Time Adaptive RNN [6], where the time-constant of the ODE is learned as a function of the hidden state and input at any time step. It accounts for the relative importance of the input and acts as a barrier to invoking the ODE solver. Whenever the time-constant is high, it barely processes the new input, remaining essentially in the previous hidden state. Alternatively, whenever the time-constant is low, it reaches equilibrium to focus on the current input. Thus, Time Adaptive RNN allows identity gradients during training and addresses the noise attenuation problem. It outperforms the SOTA architectures with 3× storage reduction and 6× compute reduction.

- **RNNs : Back-Propagation Training Issues.** Back-Propagation Through Time (BPTT) [22, 27] trains RNN by unrolling it in the time dimension and propagating the error back in time over the entire sequence length. As a result, it leads to poor trainability due to the gradient explosion/decay phenomena. In addition, it requires storing all the intermediate hidden states, resulting in significant memory costs for very long sequences. I have developed Forward-Propagation Through Time (FPTT) [7], where at each time step, we update RNN parameters by optimizing an instantaneous risk function. This proposed risk is a regularization penalty that evolves dynamically based on previously observed losses. It converges to a stationary solution of the empirical RNN objective. Empirically FPTT outperforms BPTT on several well-known benchmark tasks with up to 10% increase in accuracy on sequential decision-making tasks and up to 10 points increase in perplexity scores for sequential language modeling. Thus, FPTT enables architectures like LSTMs to learn long-range dependencies.

- **CNNs : Low Receptive Fields.** CNNs require greater depth to generate high-level features, resulting in computationally expensive models. Convolutional operators are local and restricted in the receptive field, which increases with an increase in depth. I have explored partial differential equations (PDEs) that offer a global receptive field without the added overhead of maintaining large kernel convolutional filters. I have proposed a new feature layer, called the Global layer [8], that enforces PDE constraints on the feature maps, resulting in rich features. This layer can be embedded in a deep CNN to generate a shallower network. Thus, creating compact and computationally efficient architectures, achieving similar performance as the original network. Empirical evaluation on the CIFAR and Imagenet datasets demonstrates that architectures with global layers require $2-5$x less computational and storage budget without any loss in performance.

- **CNNs : Exploiting Spatial Interpolation.** Residual blocks are the backbones of many CNNs such as MobileNets[18] and EfficientNets[24]. I have proposed a novel interpolation scheme [9] to reduce the computational cost of any generic residual block. It decomposes the output of the block as an interpolation between features processed at a low-resolution sampling of the input features and cheaper features processed at the input resolution. We use this interpolation scheme to create Spatially Interpolated Inverted Residual block and train the interpolated variants of the MobileNetV3 and EfficientNet models. Evaluation on the Imagenet dataset show up to 40% savings in compute without any loss in performance.

2 Input Hardness Aware Architectures

We can further bridge the gap between very low and very high complexity models by leveraging input hardness, namely the notion that not all input instances are equally hard to predict for a DNN. For example, very clean and centered images with brightly lit backgrounds, well-aligned with the training distribution, should be easily handled by low-capacity CNNs. Further, a high-capacity network should only execute a significantly less time for correct prediction. In contrast, the model should spend more resources on hard images such as uneven lighting or drastic distribution shift. Below, I have summarized my previous work, various novel architectures, and algorithms that leverage the input hardness to improve the performance and resource-usage trade-off.

- **Abstaining Classifier.** Selective Classification (SC) [15, 20] enables a DNN to make a prediction only when the network is confident. It allows the model to abstain from a prediction on uncertain inputs. Traditional SC methods use post-hoc strategies, such as entropy threshold on the predictive distribution, to create abstaining classifiers from standard DNNs. I have integrated the abstention mechanism within the DNN training by learning a collection of class-wise decoupled one-sided empirical risks. It finds the largest decision sets for each class with few false positives. This One-Sided Prediction (OSP)[2, 1] based relaxation yields an SC scheme that attains a near-optimal trade-off between abstention and accuracy. On CIFAR, OSP achieves 98% accuracy by predicting on 81% inputs, a 6% increase in coverage over best method.

- **Hybrid Models.** An abstaining classifier can be further specialized into a hybrid model[4, 5]. In this meta-architecture, a low-capacity base model handles easy examples and abstains from difficult ones. These inputs are routed to a specialized high-capacity global model. This hybrid design achieves SOTA accuracy and significantly outperforms the existing efficient low-complexity models. It is flexible enough to be deployed in different hybrid setups. For instance, edge-cloud setup (base on the edge and global on the cloud) or the same hardware as long as the high-capacity model can run on this hardware. This setup achieves cloud accuracy on micro-controllers with 25% latency reduction and on mobile phones with 70% latency reduction.

- **Distilling Selective Knowledge.** Further, I have leveraged this notion of input hardness in improving Knowledge Distillation (KD), a procedure widely used to distill information from a high-capacity teacher into a low-capacity student. It blindly asks the student to follow the teacher's softened probability distribution. Instead, when the student's capacity is much lower than the teacher's, it should follow the teacher only on the inputs realizable by the student function class. Informally, not all inputs are equally difficult for the student to learn, and it would benefit by focusing its capacity on the easier region. My improved KD procedure, Distilling Selective Knowledge (DiSK)[3], during training, introduced a guide function $g(\mathbf{x})$ that serves as a signal if the teacher perceives the input $\mathbf{x}$ as easy or difficult for the student, i.e.,

  - *if* $g(\mathbf{x}) \approx 1$, teacher discounts the input $\mathbf{x}$ in the student training.
  - *if* $g(\mathbf{x}) \approx 0$, teacher signals the input $\mathbf{x}$ as learnable by student.

  DiSK yields large gains over vanilla KD. It produces significantly smaller complexity students with near-teacher accuracy (e.g. 8× compute reduction with $\sim 2\%$ accuracy loss on CIFAR-100).

3 Future Research Directions

Before discussing future research work, let us compare GPT-3, a SOTA DNN, and the human brain. [19] notes that GPT-3 has twice the number of neurons compared to the human brain, yet, it can only solve language processing tasks compared to the myriad tasks performed by the human brain (audio, video, language, sentiment, creativity, etc.). In addition, training this network alone requires 50× more energy than that consumed by an average human over the whole lifespan. Thus, current SOTA DNNs are inefficient, unspecialized, and underperforming compared to the human brain. It should put in perspective how far behind the field is in the efficiency and specialization of the DNNs. My future research directions aim directly at improving this efficiency and specialization viewpoint. Below, I have listed some concrete ideas that tackle this issue.

- **Efficient Transformers.** Transformers[26] have emerged as a one-stop solution for many learning tasks in language, vision, and speech domains. They evolved from their simple siblings like RNNs, and do not inherit long-range dependency learning issues. But, this improvement comes with a set of challenges: (a) large model size and inference complexity as it requires access to dependency calculation that scales quadratically in sequence length, and (b) long training times and high working memory. My work on low-complexity RNNs can be extended to generate efficient transformers with significantly less resource consumption. In addition, the FPTT algorithm for RNNs can be modified to reduce transformer training time. Similarly, input hardness can be incorporated into the transformer architectures by adaptively invoking a low-capacity transformer for easy inputs and routing to a high-capacity transformer for hard inputs.

- **Context Aware DNNs.** Traditional DNNs rely on a single feature generation pipeline that focuses on the input and outputs the predictions. In addition to the input, the human brain relies on tangential observations to infer the context, speeding up the inference and predictive performance. DNNs should follow a similar strategy for the low-complexity and input hardness-aware architectures. Side information should help improve the DNN generalization. In the visual or textual domain, a context can be as simple as the prominent feature locations, objects, or actor information. It will enable a DNN to parse useful features before delving into the details or effectively censoring unnecessary information. I believe some logic that efficiently infers contextual information will benefit any neural architecture.

- **Enforcing Constraints During Training.** Recent works have proposed various characteristic quantities to qualify the generalization properties of a minimum of the DNN loss surface. These include Lipschitz smoothness, surface width, predictive entropy, discount on hard-to-learn data points, etc. In the literature, these metrics are used post-hoc to measure the generalization error. Instead, these quantities should be treated as constraints to be enforced systemically during training. My work on selective distillation (DiSK) proposes a simple primal-dual scheme that enforces and relaxes the constraint during training. It promotes the DNN to learn simpler hypotheses at the beginning of the training, and towards the end, it learns harder data points such that these are consistent with the most promising simple hypothesis. Thus, I believe constraints associated with better generalization properties should be enforced during training to yield efficient DNNs.

- **Concoction of Specialized Low-Capacity DNNs.** Input hardness-aware architectures can be extended to include many specialized low-capacity models that coordinate with a few high-capacity networks. This integration yields fast inference where a low-capacity model specializes in a part of the input space or a different modality. For further refinement, send low-confidence predictions to other specialized or high-capacity models. This meta-architecture can be seen as a pseudo-brain interface where different expert models (with varying degrees of capacity) handle specialized functions. It promotes resource usage reduction by sharing latent space for low-capacity models, from which they specialize into their respective functions. These specialized functions and shared latent space allow easy interplay between various modalities such as vision, language, speech, and sensory. It is in stark contrast to the mixture of experts paradigm where either all the experts share the same capacity structure or their integration is post-hoc and does not incorporate the input hardness as I have discussed earlier.

- **Applications.** I envision deploying my research to applications serving a large user base. As a Research Fellow at Microsoft Research, I deployed our Extreme Classification work [12, 11] in the Bing Ads recommendation engine, yielding significant improvements in click-through rates. This experience, alongside my graduate research work, gives me a unique perspective on application-driven research. For instance, in many mobile applications, such as Android/iOS Camera, Alexa/Google Assistant, Oculus, etc., we can deploy abstaining and hybrid models. In this setup, the low-capacity abstaining model resides on the edge device and infers as many queries as it can confidently. Once the edge model decides the uncertainty about user input, it can provide the user with its low-confident prediction and an option for the user to consent to the cloud model in case the user wants high-quality predictions. Similarly, the proposed low-complexity architectures improve computational and storage requirements for all resource constraints. Thus, they can be easily scaled and deployed in existing machine learning applications.

References by Applicant

[1] Aditya Gangrade, Anil Kag, Ashok Cutkosky, and Venkatesh Saligrama. "Online Selective Classification with Limited Feedback". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan. Vol. 34. Curran Associates, Inc., 2021, pp. 14529–14541. URL: https://proceedings.neurips.cc/paper/2021/file/79b6245ff93841eb8c120cec9bf8be14-Paper.pdf.

[2] Aditya Gangrade, Anil Kag, and Venkatesh Saligrama. "Selective Classification via One-Sided Prediction". In: *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*. Ed. by Arindam Banerjee and Kenji Fukumizu. Vol. 130. Proceedings of Machine Learning Research. PMLR, 13–15 Apr 2021, pp. 2179–2187. URL: https://proceedings.mlr.press/v130/gangrade21a.html.

[3] Anil Kag, Durmus Emre Alp Acar, Aditya Gangrade, and Venkatesh Saligrama. "Scaffolding a Student to Instill Knowledge". In: *Submitted to The Eleventh International Conference on Learning Representations*. under review. 2023. URL: https://openreview.net/forum?id=N4K5ck-BTT.

[4] Anil Kag, Igor Fedorov, Aditya Gangrade, Paul Whatmough, and Venkatesh Saligrama. "Achieving High TinyML Accuracy through Selective Cloud Interactions". In: *DyNN workshop at the 39th International Conference on Machine Learning*. 2022. URL: https://dynn-icml2022.github.io/spapers/paper_2.pdf.

[5] Anil Kag, Igor Fedorov, Aditya Gangrade, Paul Whatmough, and Venkatesh Saligrama. "Efficient Edge Inference by Selective Query". In: *Submitted to The Eleventh International Conference on Learning Representations*. under review. 2023. URL: https://openreview.net/forum?id=jpR98ZdIm2q.

[6] Anil Kag and Venkatesh Saligrama. "Time Adaptive Recurrent Neural Network". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 15149–15158.

[7] Anil Kag and Venkatesh Saligrama. "Training Recurrent Neural Networks via Forward Propagation Through Time". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 18–24 Jul 2021, pp. 5189–5200. URL: https://proceedings.mlr.press/v139/kag21a.html.

[8] Anil Kag and Venkatesh Saligrama. "Condensing CNNs With Partial Differential Equations". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2022, pp. 610–619.

[9] Anil Kag, Gourav Wadhwa, Venkatesh Saligrama, and Prateek Jain. "Spatially Interpolated Inverted Residual Block". In: under review. 2023.

[10] Anil Kag, Ziming Zhang, and Venkatesh Saligrama. "RNNs Incrementally Evolving on an Equilibrium Manifold: A Panacea for Vanishing and Exploding Gradients?" In: *International Conference on Learning Representations*. 2020. URL: https://openreview.net/forum?id=HylpqA4FwS.

[11] Yashoteja Prabhu, Anil Kag, Shilpa Gopinath, Kunal Dahiya, Shrutendra Harsola, Rahul Agrawal, and Manik Varma. "Extreme Multi-Label Learning with Label Features for Warm-Start Tagging, Ranking & Recommendation". In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. WSDM '18. Marina Del Rey, CA, USA: Association for Computing Machinery, 2018, pp. 441–449. ISBN: 9781450355810. DOI: 10.1145/3159652.3159660. URL: https://doi.org/10.1145/3159652.3159660.

[12] Yashoteja Prabhu, Anil Kag, Shrutendra Harsola, Rahul Agrawal, and Manik Varma. "Parabel: Partitioned Label Trees for Extreme Classification with Application to Dynamic Search Advertising". In: *Proceedings of the 2018 World Wide Web Conference*. WWW '18. Lyon, France: International World Wide Web Conferences Steering Committee, 2018, pp. 993–1002. ISBN: 9781450356398. DOI: 10.1145/3178876.3185998. URL: https://doi.org/10.1145/3178876.3185998.

Other References

[13] Tom Brown et al. "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[14] Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. "Once for All: Train One Network and Specialize it for Efficient Deployment". In: *International Conference on Learning Representations*. 2020. URL: https://arxiv.org/pdf/1908.09791.pdf.

[15] Yonatan Geifman and Ran El-Yaniv. "SelectiveNet: A Deep Neural Network with an Integrated Reject Option". In: *International Conference on Machine Learning*. 2019, pp. 2151–2159.

[16] K. He, X. Zhang, S. Ren, and J. Sun. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[17] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[18] Andrew Howard et al. "Searching for MobileNetV3". In: *CoRR* abs/1905.02244 (2019). arXiv: 1905.02244. URL: http://arxiv.org/abs/1905.02244.

[19] Gao Huang. *Spatially and Temporally Adaptive Neural Networks*. 2022. URL: https://icml.cc/virtual/2022/workshop/13451#wse-detail-19404.

[20] Ziyin Liu, Zhikang Wang, Paul Pu Liang, Russ R Salakhutdinov, Louis-Philippe Morency, and Masahito Ueda. "Deep Gamblers: Learning to Abstain with Portfolio Theory". In: *Advances in Neural Information Processing Systems*. 2019, pp. 10622–10632.

[21] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. "On the difficulty of training recurrent neural networks". In: *International Conference on Machine Learning*. 2013, pp. 1310–1318.

[22] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning Internal Representations by Error Propagation". In: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. Cambridge, MA, USA: MIT Press, 1986, pp. 318–362. ISBN: 026268053X.

[23] Chitwan Saharia et al. *Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding*. 2022. DOI: 10.48550/ARXIV.2205.11487. URL: https://arxiv.org/abs/2205.11487.

[24] Mingxing Tan and Quoc Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. Long Beach, California, USA: PMLR, Sept. 2019, pp. 6105–6114. URL: http://proceedings.mlr.press/v97/tan19a.html.

[25] Mingxing Tan and Quoc Le. “EfficientNetV2: Smaller Models and Faster Training”. In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 18–24 Jul 2021, pp. 10096–10106. URL: https://proceedings.mlr.press/v139/tan21a.html.

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[27] P. J. Werbos. “Backpropagation through time: what it does and how to do it”. In: *Proceedings of the IEEE* 78.10 (1990), pp. 1550–1560. DOI: 10.1109/5.58337.