Continual learning with conditional computation networks

Abstract for the general public

Over the past decade, deep neural networks have seen significant advancements driven by increasing model size and training data. Research on scaling laws suggests that further progress will require even bigger models and more training data. However, training and hosting these large networks demand substantial computational resources, which makes it costly and restricts research on state-of-the-art models to a few best-funded companies. The environmental costs of large models are also quickly becoming concerning. Machine learning-related carbon emissions and energy consumption already account for around 1of the global total in both metrics, which is comparable to medium-sized countries. Daily water consumption for cooling servers for systems like ChatGPT is estimated at 0.5 to 2.5 megaliters. Therefore, resource-saving machine learning techniques for training and inference that effectively utilize the available data are crucial to ensure the sustainable growth of the field.

Various techniques that leverage redundancy and over-parametrization of neural networks have emerged to reduce inference costs. Such techniques include quantization, pruning, knowledge distillation, and conditional computation networks. Conditional computation methods adapt the processing path in the model to the input data, intending to save computation on easier data samples by using a subset of the network. While such adaptivity is desirable, standard neural networks lack this ability and always perform the same operations regardless of input. Popular conditional computation solutions include Mixture-of-Experts, which scales up the capacity of the model by splitting it into expert modules, and early-exits, which add auxiliary classifiers to different layers of the network and allow it to dynamically skip unnecessary computation during the inference.

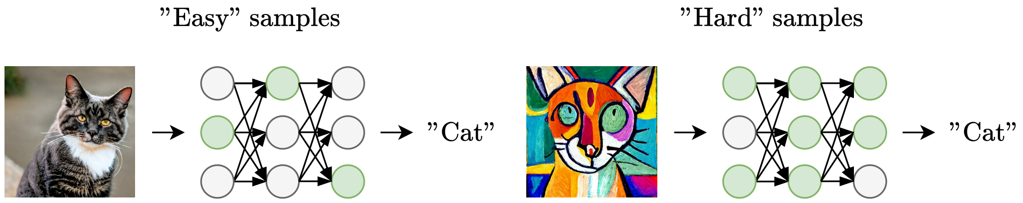

Figure 1: Conditional computation methods aim to leverage the varying difficulty of data and introduce a mechanism that allows the networks to save computation by using fewer operations for easier samples.

Techniques focusing on training efficiency have been explored in parallel. Continual learning explores learning from incremental data streams, allowing the incorporation of new data into models without forgetting previously learned knowledge. This is especially important when retraining from scratch is costly or the original training data is no longer accessible. While most work on continual learning mentions efficiency as a desirable property of their methods, in practice they prioritize state-of-the-art results, disregarding the computational costs. We believe that computational efficiency is vital for continual learning in real-world scenarios, and continual learning methods should be designed to be efficient. We propose to leverage the adaptability and compute-saving capabilities of conditional computation methods in continual learning and use early-exit architectures to obtain novel, well-performing, and efficient solutions for continual learning.

To understand the performance of early-exit networks in continual learning, we will investigate representations in those networks and measure the specialization of the individual early-exit classifiers. We will tailor early-exit networks to continual learning scenarios through mitigating task-recency bias and customized knowledge distillation. Furthermore, we will explore architectural modifications and alternative exit mechanisms for the continual learning scenarios. We will combine our findings to obtain an effective and computationally light-weight continual learning method.