Summary/ Critique
of
PArtNNer: Platform-Agnostic Adaptive Edge-Cloud DNN Partitioning for Minimizing End-to-End Latency

- **A summary of the paper:**
  The paper aims to reduce the end-to-end inference latency of Deep Neural Network (DNN) applications in edge-cloud environments by dynamically selecting the optimal partition point between the edge device and the cloud server. Traditional edge-only or cloud-only inference methods often struggle due to limited edge resources or unstable network conditions. To address this, the authors propose PArtNNer, a platform-agnostic, adaptive heuristic algorithm that eliminates the need for platform-specific profiling and offers a scalable, real-time approach to DNN partitioning. PArtNNer relies solely on runtime measurements of end-to-end latency collected on the edge device and dynamically adjusts the partition point based on current network bandwidth, cloud server load, and edge hardware performance. The algorithm incorporates an intelligent feedback loop and exploration mechanism to identify near-optimal partitioning strategies under varying conditions. It was evaluated across 15 edge-cloud system configurations, 6 popular DNN models (including ResNet101 and MobileNetV2), and 3 communication standards (Wi-Fi5, Wi-Fi6, and 5G). Experimental results show that PArtNNer achieved up to 21.1× lower latency than edge-only execution and 6.7× lower latency than cloud-only inference, closely tracking the performance of an oracle system while converging to optimal partition points up to 17.6× faster than exhaustive search. Its platform and model agnosticism, combined with comprehensive evaluation, validate that adaptive partitioning without offline profiling is a feasible, scalable, and effective solution for real-world edge intelligence deployment.

- **Main strengths of the paper (Some aspects to consider are novelty (how novel are the concepts, problems addressed, or methods introduced in the paper), clarity (is the paper well-organized and clearly written), and significance (comment on the likely impact the paper may have on the research community as a whole or on its own sub-field)):**
  The paper demonstrates several key strengths in terms of novelty, clarity, and significance. One of its most notable contributions is the introduction of a profiling-free and platform-agnostic adaptive DNN partitioning algorithm, PArtNNer, which sets it apart from prior works that rely heavily on offline characterization. Its ability to dynamically adjust partition points based solely on runtime latency measurements makes it highly suitable for real-world edge-cloud environments where hardware and network conditions vary frequently. The adaptive heuristic designed by the authors employs a smart feedback mechanism that balances exploration and exploitation, enabling the system to quickly converge to optimal or near-optimal partitioning strategies. In terms of clarity, the paper is well-structured and clearly

articulated, offering detailed algorithmic explanations, helpful diagrams, and comprehensive evaluation setups that enhance reader understanding and reproducibility. From a significance standpoint, PArtNNer shows broad applicability, validated through extensive experiments involving 15 system configurations, 6 different DNN models, and 3 wireless communication standards. Its practical utility and ability to generalize across heterogeneous platforms highlight its high potential impact on the fields of edge AI, embedded systems, and collaborative inference research.

- **Weaknesses of the paper? Some aspects to consider are novelty, clarity, and significance.** Despite its strong contributions, the paper has a few notable weaknesses related to novelty, clarity, and significance. While the adaptive partitioning method is novel in its profiling-free and platform-agnostic design, the core problem of DNN partitioning is well-established in existing literature. PArtNNer primarily builds on known concepts, and its novelty stems more from implementation practicality than from introducing a fundamentally new problem or solution paradigm. Moreover, although the heuristic design is effective, it depends on several empirically chosen parameters (such as α, k, near_idx, and part_prob), which could limit generalizability and reproducibility across applications or systems with different characteristics. There's also limited analysis of the trade-off between latency and inference accuracy, especially given that full-precision feature maps are transmitted, potentially consuming excessive bandwidth without exploring quantization or compression techniques. From a clarity standpoint, while the overall structure is solid, some algorithmic sections are complex and densely written, which could challenge readers who are less familiar with optimization heuristics. In terms of broader significance, the study is limited to a single edge-cloud setup and doesn't consider multi-edge or federated scenarios, which are increasingly relevant in real-world deployments. Addressing these limitations could further elevate the impact and applicability of the work.

- **Improvement of the study in own opinion:**
  The study could be improved in several meaningful ways to enhance both its practical utility and research impact. First, the authors could implement dynamic parameter tuning instead of relying on empirically fixed heuristic parameters (e.g., α, near_idx, part_prob). Integrating an adaptive learning mechanism or lightweight reinforcement learning strategy would help the algorithm better adjust to diverse and evolving conditions in real time. Second, the current approach transmits full-precision (FP32) feature maps, which can be bandwidth-intensive. Incorporating compression or quantization techniques (e.g., INT8 or lossy encoding) could reduce communication latency and energy consumption without sacrificing accuracy. Third, while the study offers a robust evaluation across various devices and DNNs, it is confined to a single edge-to-cloud communication model. Extending the framework to multi-edge, fog, or federated architectures would reflect more realistic deployment scenarios and showcase the

system's scalability. Additionally, including energy-aware partitioning alongside latency optimization especially for battery-powered edge devices would broaden the framework’s appeal. Lastly, the paper could benefit from a more user-friendly implementation, such as releasing open-source code and offering an API or dashboard for visualization, which would aid real-world adoption and reproducibility by other researchers and developers.