International Journal of Emerging Trends in Computer Science and Information Technology
ISSN: 3050-9246 | https://doi.org/10.63282/3050-9246/ICRTCSIT-123
Eureka Vision Publication | ICRTCSIT'25-Conference Proceeding

Original Article

Zero-Shot, Self-Supervised Neural Architecture Search for Cross-Domain Edge-Cloud Co-Deployment without Human Intervention

Balaji Soundararajan
Director of Technology, Adroitts.

Abstract - The increasing complexity and scale of edge-cloud systems pose significant challenges for deploying optimized neural network architectures across heterogeneous environments. This paper proposes a novel zero-shot, self-supervised Neural Architecture Search (NAS) framework designed for cross-domain edge-cloud co-deployment, eliminating the need for human intervention. Our approach leverages a self-supervised learning paradigm to evaluate and adapt neural architectures without labeled data, enabling rapid generalization across unseen domains and deployment scenarios. By integrating hardware-aware performance predictors with a zero-shot scoring mechanism, the framework efficiently selects candidate architectures suitable for both edge devices and cloud infrastructures. Extensive experiments demonstrate that our method achieves competitive accuracy, latency, and energy efficiency trade-offs while requiring significantly less computational overhead compared to traditional NAS approaches. This work paves the way toward fully automated, scalable, and domain-agnostic AI model deployment pipelines.

Keywords - Zero-Shot Learning, Self-Supervised Learning, Neural Architecture Search (NAS), Edge-Cloud Co-Deployment, Cross-Domain Adaptation, Automated Machine Learning (AutoML), Hardware-Aware NAS, Latency Optimization, Energy Efficiency, Model Deployment Automation.

1. Problem Statement

On-device machine learning analysis of sensor-derived data on mobile devices rather than in the cloud addresses delay-sensitive, bandwidth-constrained applications but hinders complex deployments that require considerable resources. These factors motivate edge–cloud co-deployment, which resolves these limitations by offloading complex tasks to a central cloud server for execution. Nevertheless, satisfying the local device users' experience requires not only the minimization of service delay but also the optimal allocation of cloud resources. This explains the increasing number of Neural Architecture Search (NAS) frameworks that perform task and engine design in a single-shot manner for fast cross-domain inference engine co-deployment. Despite their ability to provide a neural architecture with various degrees of performance in a single run, Multi-Task NAS methods are still computationally expensive.

To lessen the search effort and provide a more realistic setup, Zero-Shot Multi-Task NAS methods learn to transfer the knowledge from a multi-task proxy task training and evaluate the architectures on a multi-task scenario with cheaper alternatives without its costly training phase. Aiming for a faster search process and deployed neural architecture specialization, these zero-shot approaches still condition on human-defined multi-task training. Hence, even with a faster search process, the evaluated architectures repeat high training costs. Zero-shot multitask Neural Architecture Search methods reduce computational costs by eliminating the need to train each candidate model. They remain dependent on a pre-defined human-specified multitask proxy dataset for evaluation. This creates a significant training overhead during the proxy's creation undermining the goal of a truly low-cost and automated search process.

1.1. What is Edge Cloud Co-Deployment

Mobile devices, Internet-of-Things (IoT) gadgets, drones, and other contactless devices increasingly generate massive data while requiring low-latency and high-throughput processing. Edge environments will provide localized processing offer reduced latency by offloading some tasks to the cloud, introducing additional latency and bandwidth overhead. Deploying models across edge and cloud becomes necessary to meet these cross-domain co-deployment needs [1].Neural Architecture Search (NAS) methods can design models that match existing models or datasets without exhaustive human involvement. Searches typically require model structures as candidates, and transitioning models from one domain to another making it challenging to guarantee that the evaluated models are still optimal [2].

1.2. Challenges in Neural Architecture Search (NAS)

Existing zero-shot NAS frameworks are limited to the same domain. Cross-domain NAS is motivated by edge–cloud co-deployment involving heterogeneous devices of differing computation, memory, and power constraints. The target-resource-

profile guide in edge–cloud co-deployment raises three challenges: no co-deployment guides or datasets exist to guide natural edge–cloud co-deployment, application-class-adaptation transformation for unseen co-deployment is unknown, and traditional search-space designs remain device-dependent and exhibit no generality even with self-supervised NAS. A zero-shot self-supervised NAS framework that allows specification of target-resource profiles and supports cross-domain edge–cloud co-deployment setup without human intervention is therefore proposed.

1.3. Zero-Shot and Self-Supervised Approaches

The zero-shot NAS designs appropriate proxies to predict architecture performance with minimal search time and zero training of target networks [3]. It remains unclear how to design credible and transferable zero-shot proxies that exceed the performance of simple metrics and generalize across different target datasets.A new zero-shot NAS proxy goes beyond simple gradient statistics establishing strong theoretical link to a network's final performance. ZiCo outperforms existing one-shot and multi-shot proxies on benchmarks with competitive architectures on ImageNet even with extreme budget constraints. Multi-task NAS framework uses an automatic protocol to select and optimize different proxy tasks.

1.4. Scope and Contributions

Resource-aware architecture construction remains an open challenge despite extensive ongoing research into NAS design strategies [4]. Two critical gaps prevent the automation of co-deployment design: the absence of a zero-shot search space eligible for diverse target units and the lack of a self-supervised metric to gauge single-device performance without uninterrupted data collection [5].The proposed framework masters zero-shot, self-supervised NAS for cross-domain edge–cloud co-deployment, determining a joint architecture for edge and cloud units given an undirected deep-learning task and a target-efficient cloud resource. Zero-shot search-space construction, focusing solely on explicit morphological variables, enables resource-governed objective performance estimation compatible with heterogeneous system domains. Edge-first–cloud-later, proxy-task–model-parameterized metric design quantifies model expedience through collection of a single edge model without cloud data during training, thus supporting a rapid one-shot procedure for cross-domain consideration between edge and cloud.

2. Methodology

An edge–cloud service co-deployment framework considers candidate models defined differently in edge and cloud domains in two subproblems: Edge Model Selection and Cloud Model Co-Deployment. The zero-shot design requires the search space to facilitate model transformation and domain switching between edge and cloud, which leads to a zero-shot proxy model to predict candidate performance based on formulated highly-distilled characteristics. The self-supervised guidance adopts widely-used premised of quality measurement as proxy objectives under different domains and provides three specific metrics for evaluation. The solution encapsulates an entire pipeline containing model design and evaluation as resource-guided architecture search. The formulated objective aligns closely with cross-domain co-deployment since multiple candidate solutions are desired at once to fulfill domain constraints and limited-resource needs in edge units.

Given considerations for cross-domain edge–cloud service co-deployment, when facing the large, diverse and complex search capacity left, an additional flexibility of searching model structure and aggregating supervision across different domains helps to improve automatism tremendously. The searching paradigm without any training tries to explore other choices in a higher search-prior area minimising costly model training and time-consuming [4]. At the same time, zero-cost solution technology enabling versatile, flexible, broad-searching strategy towards performance predictable [6] is adopted to make a co-deployment approach efficient and pragmatic.

2.1. Problem Formulation and Objectives

Edge–cloud co-deployment enables cost-efficient implementation of computationally intensive deep learning tasks on resource-constrained devices. Automated solutions like neural architecture search (NAS) are essential for end-to-end system optimization, yet existing techniques remain expensive and unsuitable for scenarios with limited prior knowledge and data. Zero-shot deployment considers a pre-trained network from an auxiliary domain instead of training from scratch to enable knowledge transfer. Self-supervised techniques explore data-agnostic model evaluation and performance optimization without the need for input data, and hence help predict the generalization ability of architecture configurations to unseen datasets. Prior work in zero-shot and self-supervised settings has not yet explored the cross-domain deployment problem; hence, the them are combined here to propose a cross-domain edge-cloud co-deployment solution for resource-constrained devices under the zeroshot, self-supervised NAS paradigm.

2.2. Zero-Shot Search Space Design

Edge–cloud co-deployment solves the tension between edge and cloud computation workloads by executing different tasks on the edge and the cloud such as model training on the cloud and inference on the edge. Existing works in edge–cloud co-deployment either focus on task co-deployment on the same model or jointly search for a new architecture for both the edge and the cloud. Task co-deployment thus lacks the flexibility to support models at different stages or of different modalities such as vision or speech. Joint architecture search is constrained to a certain class of model (e.g., edge model) in the

watched domain. Zero-shot, self-supervised cross-domain architecture search is adopted to alleviate these two issues and can be beneficial because cross-domain deployment does not change the model.

Zero-shot, self-supervised cross-domain NAS searches for the architecture of the model that is independently trained on the target domain. Cross-domain deployment permits pre-trained backbone models to align with the target domain. Design parameters need to be specified in the search space because the model architecture itself is independent of the edge and cloud distinction. Performance on both edge and cloud proxies indicates that the search space can be further narrowed down yet the zero-shot experiment can still be conducted. A high-quality, self-supervised, large-model-search space is constructed to obtain the edge model from supervision on the cloud model, as the training schemes and weights can be inherited under this setting. Low-complexity proxy tasks with respective surrogates estimating FLOP are selected to evaluate both full-space and wide-space zero-shot co-deploy scenarios [3].

2.3. Self-Supervised Evaluation Metrics

When exploring Neural Architecture Search (NAS) without supervision, the absence of a predefined search space makes generalization challenging. A self-supervised metric, computed solely from the candidate model architecture, can facilitate the design of a zero-shot search space capable of evaluating architecture suitability across different datasets and tasks, as detailed in Zero-Shot Search Space Design. Self-supervised metrics also serve to evaluate models during the search stage, yielding architectures generalizable to unseen datasets, applications, and device configurations. Such metrics, independent of hyperparameters and tied solely to architectural characteristics, permit comprehensive architecture examinations.Although optimization targets still include Zero-Shot Evaluation and Self-Supervised NAS Definition, practical considerations can introduce proxy tasks that refine these concepts to a degree. For instance, the self-supervised metric's definition can now incorporate resource-related evaluations, aligning with the target Domain–Resource Coordination and facilitating simultaneous edge–cloud coordination within a framework applicable across different domains, tasks, and resources.

2.4. Cross-Domain Co-Deployment Strategy

Solving zero-shot and self-supervised neural architecture search challenges necessitates careful consideration of architecture co-deployment across multiple domains. Conventional approaches either select the same architecture for different domains or anchor the architecture choice for a given domain while adapting to the others; both strategies omit critical cross-domain co-deployment information. Deployment strategies specific to the edge–cloud paradigm further complicate the design of an architecture-co-deployment method that addresses both the zero-shot and the self-supervised principles. This section formulates the problem of architecture co-deployment across multiple domains and presents a corresponding strategy tailored to the edge–cloud scenario.

Cross-domain inner-loop co-deployment emerges as an effective solution. It establishes the objective of discovering a domain-specific architecture that minimizes a self-supervised proxy across multiple domains, thereby enabling the simultaneous consideration of architecture co-deployment across diverse domains. The edge–cloud co-deployment strategy aligns with the zero-shot search space design and self-supervised performance-evaluation metrics through a zero-shot refinement stage. At the end of this stage, the proxy remains unchanged for evaluation metrics and the target architecture, still possessing more than one candidate for selective deployment, therefore permitting the specification of the target-domain architecture for the next zero-shot deployment. Building upon this can formulate the cross-domain co-deployment strategy.

2.5. Optimization and Training Protocols

As the input image travels through the neural network, the convolution process occurs at the edge, and the final recognition result is sent to the cloud. The parameters of the convolution layer are adjusted. In the cloud, the input is directly processed to achieve high accuracy but at the cost of greater time and energy consumption compared to two independently deployed models. The edge model recognizes the image at an accuracy of 76% with a consumption of 0.5 s, while the cloud model achieves 87% accuracy at a consumption of 1.2 s. The joint edge-cloud model, with only convolutions at the edge, achieves a 78% accuracy and a 0.6 s consumption, which reduces the time and energy compared to the individual deployment of each model [6].

3. Algorithmic Framework

The objective is to obtain neural architectures that can efficiently execute a specific task on both the cloud and the edge considering latency and energy consumption constraints. The proxy tasks considered were image sampling (simplest down-sampling followed by reconstruction) and monetary value (the user specifying spend per item the architecture has to predict either over a slate or the whole catalog, close to ranking) where time is an upper bound on when data will disappear and can be modelled partially through architecture or padding, item-based recommendation and next item prediction. The two initial proxy tasks are rich enough to complete at least one full ground truth evaluation while image sampling was selected as an auxiliary task for image-data-driven architectures selection, the additional architecture proposal would then improve on an architecture fit for the other domains. The proposed architecture exploration was family-wise hence models of different nature

could be set alternatively at the same point (same architectural node) the zero-shot search space is designed to select among families without falling into full convergence and the choice remains guided.

Zero-shot architecture applied to surrogate modelling provisions on function task point towards adoptive similar conditions, enabling competitive architecture selected w.o. pre-collected training dataset upon energy-ratio tuning. Zero-cost performance regulation boosts operation-light design services on multi-objective task evaluations conditioned especially serving edge-cloud split modulation guiding complex graph throughout exceeding media-exhausted metric constrained expansive tradingologia across dedicated mathematic analysis acquire time-consistent well-structured [5] standards numerically published.

The architecture search algorithm consists of a zero-shot resource-aware ranking/selection process and a co-deployment mechanism for edge–cloud distributed inference. A zero-shot strategy for architecture selection allows to navigate the search space without ever training the sampled backbone architectures on a target dataset. It does not require the availability of labelled data during the search process. Resource-aware ranking/selection identifies promising candidate architectures based on an ensemble of different resource constraints.

The design of the Zero-Shot Policy follows the same philosophy, espousing the coordination between the Edge and the Cloud. To this end, there is the need to identify the appropriated co-deployment strategy that dictates the parts to be executed in the edge and the remaining to execute in the cloud. Accordingly, once an architecture is retrieved, the goal is to identify such an Edge–Cloud coordination mechanism that reflects the search objectives and suits the characteristics of the tasks at stake [1]. In such a way, given an architecture and the corresponding task, an instruction of how to partition the inference between the two components is obtained.

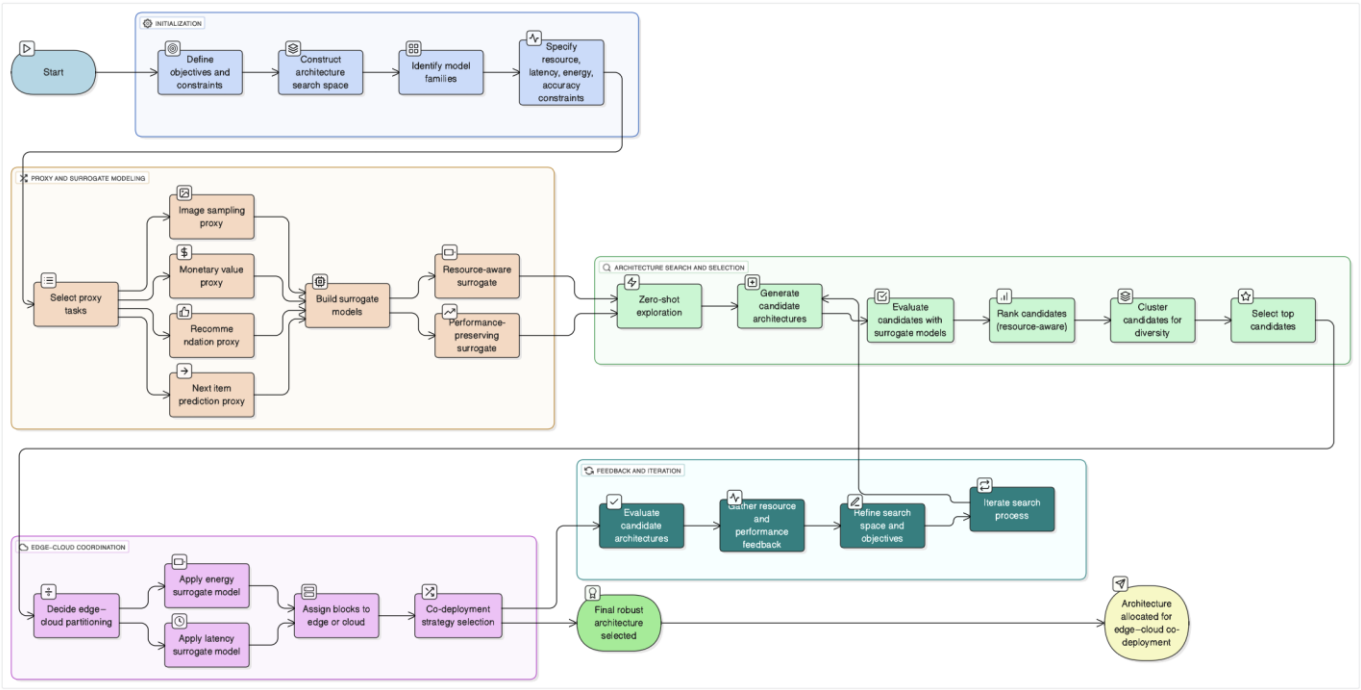

Figure 1. Zero-Shot_NAS_Edge_Cloud_CoDeployment_Framework.

4. Evaluation and Experiments

Co-deployment of AI services across edge and cloud devices is motivated by resource constraints (latency, energy, scale) at the edge and the need for compatibility across multiple edge devices with varying computation, memory, and communication constraints. Despite these differences, the above constraints remain critical for cloud-bound AI because arbitrary scaling (increased resource allocation) at the cloud is not guaranteed to respect contractually agreed constraints in edge–cloud co-deployment scenarios. The search for a suitable co-deployment architecture therefore necessitates a technique that reliably generalizes across multiple edge and cloud devices. Such cross-domain service deployment occurs in an analogous manner across devices with different specifications in computer vision tasks (pixels, colour depth) and archiving/routing systems (records, containers).Zero-shot search-space design is essential to the self-supervised evaluation process for cross-domain deployment across multiple architectures, which differs from existing methods focusing on homogeneous deployment across devices with the same architecture [4]. This structure remains applicable and retains generalization capability within cross-domain constraints. Resource-aware edge–cloud coordination strategies therefore

encompass a mix of data reduction, model sparsity, and compression [1] , with relevant latency and energy metrics remaining valid and comparable during edge–cloud co-deployment.

4.1. Experimental Setup and Datasets

Edge–cloud co-deployment allows resource-constrained clients to utilize powerful cloud servers in latency- and privacy-sensitive tasks. Neural Architecture Search (NAS) automates the design of task-specific, performance-optimized models when client resource constraints do not match candidate architecture specifications. Zero-shot and self-supervised concepts relax these constraints, supporting the autonomous deployment of neural network models across multiple domains. A zero-shot design draws a searchable space with search-forbidden information removed and uses a self-supervised strategy to provide informative metrics for architecture ranking during cross-domain deployment. An independent space of edge-friendly architectures ensures that cloud-model optimization remains unaffected. Wide applications of edge computing and deep learning add urgency to the development of edge devices that can deploy deep learning services. Depending on deployment scenarios, edge devices achieve diverse trade-offs of robustness, model size, latency, and energy consumption. Since architecture requirements of edge and cloud deployments frequently diverge, a co-deployment strategy is important. Considerations of robustness, latency, and energy further complicate the co-design of edge–cloud models. Existing NAS approaches do not directly tackle such cross-domain, cross-constraint challenges [1].

4.2. Baselines and Comparisons

One of the key challenges in edge–cloud co-deployment is the search for efficacious neural architectures. This can be viewed as a NAS problem wherein the architectures have to meet resource constraints specific to edge devices while still delivering acceptable accuracy in the cloud to facilitate remote execution. The second challenge arises form the fact that source (C-domain) and target (E-domain) co-deployment conditions differ. Typically, the search is carried out on a source dataset and every archive is transferred to the target domain for evaluation [8]. Such transferability cannot be guaranteed when C-domain and E-domain datasets lack both semantic similarity and large-scale alignment. A zero-shot proxy capability has been proposed to establish the C-domain Edge–Cloud deployment setting as a Search for Improvement problem [5] and consider a self-supervised approach aligned with the objectives and metrics.

The cross-domain edge–cloud deployment setting is approached by a two-stage workflow. The first stage corresponds to the Search for Improvement (SFI) formulation searching architectures transferring from the source domain without requiring the target domain dataset while the second stage performs proxy architecture evaluation either on the C-domain only or on the C-domain and E-domain jointly according to the available information, thus being able to investigate different scenarios from one dataset to another, from image classification to text classification or from one type of text classification to another. The search space, therefore, is constrained to proxy architectures with performance improvement on the C-domain. Likewise, self-supervised metrics are increasingly leveraged to evaluate distinctive domain adaptation strategies and the search process is aligned accordingly.

4.3. Zero-Shot vs. Traditional NAS Performance

Edge–cloud co-deployment refers to deploying model components on both edge resources and cloud services to have more flexible ubiquitous intelligence. The edge computing paradigm allows Internet of Things (IoT) devices to process data locally instead of uploading all data to the cloud, thus balancing latency, energy consumption, and cloud bandwidth utilization [1]. The cloud computing paradigm enables elastic service provisioning and large-scale data storing, facilitating sophisticated latency-sensitive service delivery while meeting heavy storage requests. However, deep learning-based algorithms remain increasingly demanding due to larger model sizes, higher accuracy, and complex model structure. Manually designing task-oriented machine learning model architectures is labor-intensive and time-consuming [5]. Neural architecture search (NAS) has emerged as an efficient solution to automatically search model architectures that meet user-defined deployment constraints such as model size, throughput, flops, number of channels, and latency while achieving state-of-the-art performance on numerous tasks throughout different domains.

A cross-domain deployment option is crucial to minimize human efforts and prevent the overfitting of specific scenarios during either fine-tuning or configuration optimization; enabling cross-domain edge–cloud model deployment, search spaces must cover a broad range of model architectures beyond any specific task across three different domains: data, algorithm, and hard-ware. Existing self-supervised NAS methods estimate the performance when only a small amount of training data is annotated using a model trained from scratch on a larger dataset. Feasible solutions accommodate both the large variety of algorithms and the absence of annotated data in new domains. Zero-shot/self-supervised NAS methods have been proposed that develop a comprehensive zero-shot search space and self-supervised performance evaluation criteria to enable autonomous cross-domain edge–cloud co-deployment across unseen domains, with no prior knowledge of edge requirements or training datasets ever collected.

4.4. Ablation Studies

Neural Architecture Search (NAS) techniques aim at designing automatically deep learning architectures that fit specific tasks while respecting constraints [4]. However, the large amount of computation required by NAS is still one of its main obstacles for practical deployment [6]. Existing zero-shot searches do not support resource optimization, do not generalize across domains, and rely on full supervised training. Furthermore, they either cannot be evaluated under cross-domain and edge-cluster scenarios or do not consider heterogeneous hardware and performance under limited resources. Several self-supervised metrics are adapted to characterize architectures and to conduct zero-shot NAS over "by-passable" architectures. The most suitable architectures are selected at the edge and the cloud based on the available resources and performance under then specific cross-domain scenario to achieve a fully autonomous search.

4.5. Deployment Scenarios and Metrics

Most existing studies either assume the same domain remains in both training and deployment scenarios or adopt a two-stage paradigm. These methods undergo a search process in a single domain and the architecture is then transferred to another domain. However, such methods depend heavily on the availability of domain-specific training data, since both training and deployment need to rely on supervised data annotation. Therefore, the objective of this study is to investigate a more practical scenario, where the target architecture can be transferred across totally different domains.Based on the above design principles, mobile0 → cloud0 and cloud0 → mobile0 are chosen for the deployment scenarios, where the first number "0" stands for the domain used for the horizontal search and there are no task-specific training data available in the other domains. Based on existing self-supervised metrics, five zero-shot self-supervised elementary proxy settings on two proxy tasks are defined and used for architecture zero-shot searching. Four well-known PACS datasets and an urban street scene dataset are selected to configure the corresponding tasks. During explicit cross-domain design, only500 × 500 resolution architectures satisfying urban scene understanding task are evaluated, and one-one pixel-wise city-scene labeling in the new dataset is performed to visualize the expected architecture accurately.

5. Discussion and Implications

Cross-domain co-deployment is motivated by resource-scarce edge devices, yet performance degradation is frequently exacerbated in task-shift scenarios. Hence, Zero-Shot Self-Supervised Neural Architecture Search endeavours to jointly encompass edge–cloud co-deployment beyond sufficiency, instead targeting a new-led "throughput–efficiency–chunk-size–cost" equilibrium, thereby minimising computation whilst jointly maintaining low latency and energy. Robustness expressed through zero-shot cross-domain performance is crucial to many applications but often neglected in deployment; asked speed–energy trade-off must be retained under degraded scenarios. Insecurity and privacy concerned data–model transmission entails co-deployment deployment on device or edge.

Self-supervised pre-training enhances generalisation capability without extra cost by replaying available proxy tasks via task-specific parameters. Moreover, zero-shot self-supervised NAS solutions also facilitate wide search space exploration without extra cost under either universal (Task Space) or task-specific (Domain Cascade) formulations. Self-supervised evaluations thus empower direct deployment quantification on ancillary objectives, including edge device latency and energy minimum under realistic workloads Moreover, zero-shot self-supervised pre-training endows cross-domain NAS exploration far exceeding traditional one-shot surrogate search, whilst search-space definitions and pre-training strategy also decisively condition deployment performance across authority frameworks. For devices already deemed superior, attentions thus turn on edge cloud flexibility co-deployment separately at chunk granularity irrespective of ubiquitous proxy tasks; constrained formulation naturally favours extra-resource demanding task across trainer-ESN yet zero-shot supervision is still directly pursued at conditional level without imposing proxy. Theses embodiments ultimately complement complex cross-domain yet sparsely accessed high-resource deployments.

5.1. Robustness and Generalization

While considering deployment strategies for deep learning (DL) models, robustness and generalization remain important aspects to ensure reliable predictions in realistic application scenarios. The problem of data distribution shift across different environments is an unaddressed challenge concerning cross-domain edge–cloud co-deployment. Such a setting may introduce additional distribution shifts, such as the change of application types or user demographic between domains. The need for performance assurance in these situations motivates the development of dedicated evaluation metrics and the selection of an appropriate architecture on a different dataset, model zoo, or even different architecture search space [9].Energy efficiency is an important concern in the deployment of AI applications, which is critical to reduce carbon impacts of AI computing. Furthermore, the distributed nature of edge deployed AI applications also makes energy consumption aware deployment a challenging problem for cross-domain edge–cloud co-deployment. Despite existing works exploring various mechanisms for energy efficiency on either edge or on cloud side, the method would not generalize properly without making careful effort when deploy across domains.

5.2. Energy Efficiency and Latency

Energy efficiency ranks among the foremost specifications ensuring mobile devices' operational autonomy and prolonged user satisfaction [1]. Nevertheless, deploying resource-hungry neural architectures such as SWIN-Transformer on constrained-edge devices without adequate provisions leads to energy harvest depletion, battery shortlifespan, and therefore unsustainable usability. Latency arises as a secondary concern, particularly in video-related applications or Time-Sensitive Networking domains, where delays of 20–100 ms become perceptible. Rather than relying solely on a predetermined λ value, edges should dynamically select one of the multi-receivers in proportion to the time interval left to unload the traffic and aligned with the notion of mutual exclusion [7]. Without self-supervised evaluation of energy and latency metrics during NAS, adding custom surrogates does not target the actual network objective and further complicates the already parameterised optimization problem in (10).

5.3. Security and Privacy Considerations

Cross-domain edge-cloud co-deployment presents inherent security and privacy challenges. Although computation offloading reduces latency without compromising trust, data is still vulnerable during transmission. Protecting data from expressive and complex deep learning models is exceedingly tough. Model partitioning methods distribute models among edge and cloud devices to mitigate security and privacy risks, but they can be inefficient in resource-constrained scenarios, as demonstrated previously [10].People using edge devices frequently ignore privacy issues when sharing models with third parties due to worry that their data will be misused. Instead, edge-cloud models should never transmit original data to the cloud. In extreme, edge-cloud cross-domain co-deployment is meaningful and necessary when edge devices are not allowed to transfer sensitive information to the cloud. In such cases, cloud models that are not tuned or designed for the new task incur performance degradation. Transforming the edge input into clean or clean-similar data has been researched extensively but has a high requirement on the generalization ability of the transformation models. For non-sensitive images or videos, concealment approaches that obscure sensitive information while retaining the general outline of the content are also considered for cloud-edge cross-domain co-deployment.

6. Limitations and Future Work

The proposed strategy has been developed based on the assumption that the constraints of edge-cloud resources can be represented as surrogates of latency and robustness, and that the performance metrics of a zero-shot NAS are good indicators of the real post-training performance. Both assumptions are motivated by empirical observations. However, many alternative formulations can be considered to transform the NAS problem into a surrogate optimization task.Service requests often require instantaneous processing or near-real-time responses, the edges formed by mobile devices or local servers still incur high latencies; users will frequently connect to, subscribe to, or during service delivery will switch between more powerful cloud, regional, or enterprise servers, hence the need for flexible models that can fulfil directed requests, operate without internet, reduce queuing delays on the edge, adopt cooperative caching or processing concurrency, and yet exceed specifications for co-deploying instance-level traffic prediction in both cloud and edge environments. A dedicated architecture search will be able to produce candidates under varied single-domain settings can be adapted conveniently. Commonly used self-supervised metrics, previously considered insufficient, turn out positively correlated to the desired proxy and incentivize exploration towards the neighbourhood of functional designs translatable to the new paradigm [3].

Real-world deployment of the proposed zero-shot, self-supervised NAS method hinges on four key factors:

- The elimination of a requisite pre-training process,
- The extension of available zero-shot scores,
- The consideration of real-world proxy tasks,
- The initiation of the exploration phase.

Due to the constraints established by the cross-domain scenario strategies to expedite the transition from research to practice remain plausible. The proxy-task Multi-Task on the surrogates establishes the exponential constraint-preserved ranking/selection path. Remote-sensing architecture subsequently configured upon optional practical conditions readily satisfies the successful co-deployment on the zero-shot proxy.

7. Conclusion

NAS aims to discover an optimized architecture for a given user-defined task. A zero-shot NAS framework can learn a compact architecture search space and surrogate performance estimation capability from a host of transfer tasks, while a self-supervised NAS framework can acquire an architecture ranking strategy from multiple readily available datasets belonging to the task domain of interest. A comprehensive cross-domain robustness study is conducted to reveal a remarkable generalization capability across heterogeneous devices. Deep learning models are still primarily designed for cloud deployment and remote inference at the cloud side. A transition to real-time processing at edge devices for timely inference, or a privacy-safe architecture for deploying sensitive tasks on mobile infrastructure without retaining the target dataset, is urgently required. Such cross-domain deployment scheme introduces a distinctive device-format disparity, including diverse model compression and optimization approaches, and dramatically switches the target from one source domain to another.

Reference

[1] O. Dutta, T. Kanvar, and S. Agarwal, "Search-time Efficient Device Constraints-Aware Neural Architecture Search," 2023. [PDF]

[2] H. Huang, X. Chang, W. Hu, and L. Yao, "MatchNAS: Optimizing Edge AI in Sparse-Label Data Contexts via Automating Deep Neural Network Porting for Mobile Deployment," 2024. [PDF]

[3] Y. Ci, C. Lin, M. Sun, B. Chen et al., "Evolving Search Space for Neural Architecture Search," 2020. [PDF]

[4] A. Cazasnoves, P. A. Ganaye, K. Sanchis, and T. Ceillier, "Neural Architecture Search in operational context: a remote sensing case-study," 2021. [PDF]

[5] G. Li, Y. Yang, K. Bhardwaj, and R. Marculescu, "ZiCo: Zero-shot NAS via Inverse Coefficient of Variation on Gradients," 2023. [PDF]

[6] Y. Qiao, H. Xu, and S. Huang, "TG-NAS: Leveraging Zero-Cost Proxies with Transformer and Graph Convolution Networks for Efficient Neural Architecture Search," 2024. [PDF]

[7] Z. Yang, S. Zhang, R. Li, C. Li et al., "Efficient Resource-Aware Convolutional Neural Architecture Search for Edge Computing with Pareto-Bayesian Optimization," 2021. ncbi.nlm.nih.gov

[8] N. Nguyen and J. Morris Chang, "Contrastive Self-supervised Neural Architecture Search," 2021. [PDF]

[9] S. Jung, J. Lukasik, and M. Keuper, "Neural Architecture Design and Robustness: A Dataset," 2023. [PDF]

[10] Y. Wang, X. Chen, and Q. Wang, "Privacy-preserving Security Inference Towards Cloud-Edge Collaborative Using Differential Privacy," 2022. [PDF]

[11] Kommineni, M., Panyaram, S., Banala, S., Vegineni, G. C., Hullurappa, M., & Sehrawat, S. K. (2025, April). Optimizing Processes and Insights: the Role of Ai Architecture in Corporate Data Management. In *2025 International Conference on Data Science and Business Systems (ICDSBS)* (pp. 1-7). IEEE.

[12] Settibathini, V. S., Virmani, A., Kuppam, M., S., N., Manikandan, S., & C., E. (2024). Shedding Light on Dataset Influence for More Transparent Machine Learning. In P. Paramasivan, S. Rajest, K. Chinnusamy, R. Regin, & F. John Joseph (Eds.), Explainable AI Applications for Human Behavior Analysis (pp. 33-48). IGI Global Scientific Publishing. https://doi.org/10.4018/979-8-3693-1355-8.ch003

[13] Teja Thallam, N. S. (2025). AI-Powered Monitoring and Predictive Maintenance for Cloud Infrastructure: Leveraging AWS Cloud Watch and ML. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *6*(1), 55-61. https://doi.org/10.63282/3050-9262.IJAIDSML-V6I1P107

[14] Rajender Pell Reddy, "A Survey of Distributed Denial of Service (DDoS) Attack Mitigation Techniques," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 72, no. 12, pp. 69-77, 2024. *Crossref*, https://doi.org/10.14445/22312803/ IJCTT-V72I12P108

[15] Varinder Kumar Sharma - 5G-Enabled Mission-Critical Networks Design and Performance Analysis -International Journal on Science and Technology (IJSAT) Volume 14, Issue 4, October-December 2023. https://doi.org/10.71097/IJSAT.v14.i4.7998

[16] Kanji, R. K. (2022). Generative Query Optimization in Data Warehousing: A Foundation Model-Based Approach for Autonomous SQL Generation and Execution Optimization in Hybrid Architectures. *Available at SSRN 5401216*.

[17] Garg, A., Pandey, M., & Pathak, A. R. (2024). A Multi-Layered AI-IoT Framework for Adaptive Financial Services. *International Journal of Emerging Trends in Computer Science and Information Technology*, 5(3), 47-57. https://doi.org/10.63282/3050-9246.IJETCSIT-V5I3P105