Kern Public : Pages:16–29

Original Research

Machine Learning as a Service: Opportunities and Challenges for Big Data Processing in the Cloud

Siti Norlina Binti Zulkifli[1] and Zainuddin Bin Yusof[2]

[1]Perlis College of Computing, Department of Software Systems, Jalan Bukit Lagi No:42, Kangar, Perlis, Malaysia.
[2]Research Assistant at Malaysia University of Science and Technology.

Abstract
Machine Learning as a Service has emerged as a powerful model for facilitating scalable and on-demand analytics capabilities in the cloud, especially in the context of massive datasets generated by modern enterprises. By leveraging virtualized resources, efficient data pipelines, and sophisticated optimization techniques, service providers are able to address the computational and storage requirements associated with big data processing. However, challenges remain in ensuring robust performance across geographically distributed data centers, maintaining data security and privacy, and adapting algorithms to diverse industrial applications. Additionally, questions persist concerning the integration of heterogeneous datasets that originate from varying sources and domains, raising concerns about reliability and fairness in predictive modeling. The use of automated pipelines and containerized deployments has offered valuable advantages in terms of reproducibility and ease of management, but also introduces complexities in performance tuning and resource orchestration. Despite these obstacles, ongoing research and development efforts have led to notable advancements in deep learning architectures, parallel training strategies, and high-throughput streaming analytics. Further exploration into specialized hardware accelerators and advanced resource scheduling strategies is expected to drive the future evolution of Machine Learning as a Service, while highlighting the need for new frameworks and standards. Consequently, the promise of high-impact, real-time analytics is firmly within reach, pushing innovation in fields ranging from healthcare to finance.

1. Introduction

Machine Learning as a Service has transformed the way organizations deploy, scale, and manage predictive modeling in the era of big data [1]. It represents a paradigm shift from on-premises computation to cloud-hosted solutions that allow users to interact with sophisticated algorithms through web-based interfaces or programmatic APIs. This transition has largely been motivated by the escalating volume and diversity of data being collected worldwide, encompassing textual logs, sensor readings, video streams, and user interactions on social media platforms. Consequently, the push for rapid and accurate insights has spurred the demand for robust and flexible frameworks capable of handling massive parallelism, elasticity, and dynamic resource allocation [2]. The commercialization of data analysis pipelines has further sparked interest in designing cloud architectures that capitalize on load balancing, container technologies, and distributed storage systems, thus promoting seamless integration and efficient operation.

Foundational developments in virtualization have laid the groundwork for the proliferation of Machine Learning as a Service. The introduction of hypervisors and related container runtimes, for instance, has simplified the process of provisioning multiple tenants on shared hardware, allowing them to train and deploy predictive models in isolated environments [3]. During early phases of cloud adoption, many organizations hesitated to move critical data and core analytics workloads to off-site infrastructures. However, technological improvements in network bandwidth, along with enhanced security protocols, have mitigated these concerns. As a result, high-performance computing resources such as Graphics

Processing Units and Tensor Processing Units have become readily accessible through pay-as-you-go models, thus democratizing advanced machine learning capabilities [4]. The significance of this democratization cannot be overstated, since it directly impacts smaller entities seeking to leverage cutting-edge analytics without major capital expenditures in hardware or software expertise.

There is a growing demand for systematically exploring how big data interacts with machine learning pipelines at scale, and how novel algorithms can effectively utilize distributed computing resources. In addition, developers and data scientists wish to adopt automated pipelines that streamline model training, testing, deployment, and monitoring while reducing the need for manual intervention [5]. These pipelines, often referred to as continuous integration and deployment mechanisms, have gained popularity in the industry due to their potential for faster iteration cycles and improved collaboration among multi-disciplinary teams. Beyond deployment considerations, data preprocessing and feature engineering have become crucial components of machine learning services. Automated transformations, anomaly detection methods, and feature extraction algorithms now operate under tight latency constraints to meet real-time or near-real-time requirements across a variety of application domains. [6, 7]

Nevertheless, many challenges remain in the realm of data privacy, fairness, and algorithmic accountability, since the provisioning of machine learning in the cloud exposes inherent risks related to data governance. Regulatory frameworks attempt to address these risks, but there are inevitable trade-offs between maintaining compliance and ensuring high model accuracy and interpretability. Achieving a balance between confidentiality and utility poses intricate problems for service providers and end users alike [8]. Furthermore, data sovereignty concerns have emerged as large-scale infrastructure providers span multiple countries, each with its own legal stipulations on data handling and storage. These complexities necessitate robust encryption schemes, multi-party computation mechanisms, and reliable audit trails that can track data usage without hindering performance.

Machine Learning as a Service also intersects significantly with edge computing architectures, where small, resource-constrained devices generate continuous streams of data [9]. In many situations, pushing raw data to centralized cloud servers for analysis is neither feasible nor cost-effective, prompting the development of hybrid solutions that distribute the computation between the cloud and the edge. This integration brings forth further complexities in resource scheduling and fault tolerance but offers significant gains in responsiveness. Over time, continuous research in this area will likely uncover new ways to combine state-of-the-art deep learning architectures with edge intelligence, thus scaling analytics in diverse application scenarios. [10]

In the subsequent sections, a comprehensive discussion is presented on conceptual foundations, mathematical modeling approaches, scalability and performance issues, and security and regulatory constraints. The conclusion highlights potential growth areas, limitations in current methodologies, and avenues for ongoing research.

2. Conceptual Foundations of MLaaS

Machine Learning as a Service is predicated on the seamless amalgamation of cloud resources, big data ecosystems, and advanced analytics techniques [11]. The foundational premise lies in decoupling infrastructure management from machine learning development, thereby enabling users to focus on creating predictive models rather than provisioning and configuring hardware. This approach is underpinned by a multilayered architecture that spans data ingestion, distributed processing, orchestration, model training, and deployment. Each layer incorporates subcomponents responsible for load balancing, resource allocation, security, and workflow coordination [12]. As such, the conceptual underpinnings of MLaaS extend beyond simple hosting of algorithms to include a holistic environment designed for the continuous development and refinement of predictive capabilities.

At the data ingestion layer, structured and unstructured data streams originate from heterogeneous sources, including transactional databases, streaming platforms, Internet of Things sensors, and multimedia content repositories. These streams are collected in scalable data lakes or warehousing systems that employ distributed storage schemes for fault tolerance and quick retrieval [13]. The data is frequently subject to schema transformations and metadata annotations that facilitate subsequent processing steps. In some architectures, specialized connectors and gateways ensure reliable data transfer from on-premises sites or third-party services to the cloud-based environment. Furthermore, automated triggers might initiate data preprocessing tasks to handle missing values, normalize attributes, and detect anomalies in real time. [14, 15]

The distributed processing layer typically leverages resilient cluster frameworks to handle large-scale computations and to partition data across multiple nodes for parallel processing. This division of labor is important when training large models that might involve billions of parameters. Instead of relying on a single node, the workload is split so that multiple machines handle subsets of data or distinct computational tasks, thereby reducing total training time [16]. Parallelization is orchestrated by scheduling systems that aim to minimize idle resources and optimize throughput. The interplay of these scheduling systems with container orchestration tools adds another layer of complexity, as containerization encapsulates model runtimes and dependencies in isolated environments. This strategy allows multiple users to share underlying infrastructure while retaining separate workspaces. [17]

The orchestration layer integrates version control, experiment tracking, and automated tuning processes. For instance, hyperparameter optimization is often performed via grid search, random search, or more advanced Bayesian-based approaches that systematically probe the parameter space. By coupling these optimization routines with continuous integration pipelines, MLaaS platforms can automate the retraining of models when new data arrives or when performance metrics indicate a degradation in predictive accuracy [18]. This continuous retraining model transforms machine learning from a one-time effort into an evolving process that adapts to changing data distributions and evolving business requirements.

The deployment layer is central to the MLaaS concept, as it determines how trained models are exposed as services to end users or downstream applications. Model inference can be offered through RESTful APIs, streaming interfaces, or batch processing endpoints, with elasticity mechanisms that automatically scale the number of inference nodes up or down based on workload demands [19, 20]. These mechanisms are typically managed through integrated load balancers and service meshes, ensuring that traffic is distributed evenly among available computational resources. This approach allows MLaaS providers to deliver consistent service-level agreements for latency, throughput, and reliability, even during demand spikes.

Despite these conceptual pillars, MLaaS confronts issues related to resource overcommitment, underutilization, and cost management [21]. Because cloud resources are billed on a usage basis, effective cost control requires sophisticated monitoring and scheduling algorithms that can dynamically allocate or deallocate computing instances. Overprovisioning leads to unnecessary expenses, while underprovisioning can harm service quality and user satisfaction. Therefore, advanced workload prediction techniques and intelligent resource schedulers are essential for achieving an optimal balance between cost and performance [22]. The ongoing evolution of MLaaS also intersects with platform engineering trends, making it possible to integrate ephemeral serverless functions for lightweight inference tasks, thus further expanding the array of deployment options.

Another noteworthy aspect of the conceptual framework for MLaaS is extensibility. Users vary widely in their analytics requirements, from small-scale projects involving conventional regression models to large-scale image recognition or natural language processing tasks that employ multi-layer deep networks [23]. To accommodate these varied needs, the MLaaS architecture must remain modular and support a diverse catalog of algorithms, libraries, and frameworks. Containerized microservices serve as building blocks that can be updated or replaced without disrupting the entire pipeline, offering both flexibility and maintainability. Consequently, the conceptual foundations of MLaaS emphasize decoupled services and standardized communication protocols to allow rapid experimentation, seamless upgrades, and integration with external systems.

In practice, these foundational concepts translate into an ecosystem that not only provides raw computational power but also addresses the entire lifecycle of machine learning solutions [24]. From data collection and curation to model deployment and monitoring, the architecture ensures that analytics become integral to business operations. This ability to embed machine learning insights into operational workflows, combined with the elasticity and resilience of cloud platforms, is what truly differentiates MLaaS from traditional approaches. Nevertheless, realizing these conceptual foundations at scale calls for rigorous mathematical models, algorithmic frameworks, and sophisticated performance engineering, all of which are discussed in subsequent sections. [25, 26]

3. Mathematical Modeling and Algorithmic Frameworks

A key challenge in large-scale machine learning implementations is formalizing the learning process in a manner conducive to parallel computation and distributed optimization. Let $X \in \mathbb{R}^{n \times d}$ represent a dataset with n samples and d features. A supervised learning objective could be formulated as a loss function $\mathcal{L}$, coupled with a regularization term $R(\theta)$, over model parameters θ. The problem often takes the form

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(f(x_i;\theta), y_i) + \lambda R(\theta).$$

When n and d are large, gradient-based methods are distributed across multiple worker nodes, each computing partial gradients $\nabla \mathcal{L}_j$ on subsets of the data. These gradients are then aggregated via asynchronous or synchronous parameter servers [27]. The parameter update step often follows the form

$$\theta \leftarrow \theta - \eta \Big(\frac{1}{|B|} \sum_{i \in B} \nabla_{\theta} \mathcal{L}(f(x_i;\theta), y_i) \Big),$$

where B denotes a mini-batch of data and η is the learning rate. The design of consistent update rules in the presence of network delays, stragglers, and heterogeneous node capabilities is central to the theoretical underpinnings of distributed machine learning. [28]

Another domain of mathematical interest arises in unsupervised learning tasks, which may involve clustering or dimensionality reduction. Consider a clustering objective that employs the K-means algorithm. The partition of X into k clusters is found by iteratively minimizing [29]

$$\sum_{i=1}^{n} \min_{1 \le j \le k} \left\| x_i - \mu_j \right\|^2,$$

where μ_j is the centroid of the j-th cluster. Parallelization of K-means in MLaaS requires distributing both the data points and the centroid updates across multiple nodes. The algorithm's convergence properties and runtime performance hinge on how communication overhead and load imbalance are handled. [30, 31]

Neural networks, especially deep architectures, form another significant segment of MLaaS. The feedforward process can be thought of as a layered composition of functions

$$z^{(l+1)} = \sigma\big(W^{(l)} z^{(l)} + b^{(l)}\big),$$

where $z^{(l)}$ and $z^{(l+1)}$ denote the activations in layer l and $l+1$ respectively, $W^{(l)}$ are weights, $b^{(l)}$ are biases, and σ is a nonlinear activation function. The backpropagation step distributes gradients throughout the network to update these weights [32]. In large-scale implementations, the computational graph is split among multiple devices, each responsible for a subset of the layers or data samples. This partitioning can be formulated as a graph-partitioning problem, with the objective of minimizing inter-device communication while balancing computational load. The complexity of scheduling these computations and synchronizing parameter updates is a topic of active research, involving advanced mathematical tools like graph cut formulations and approximate solutions based on heuristics or evolutionary algorithms. [33]

In scenarios where data arrives continuously, online learning or streaming analytics frameworks are employed. The learning objective might then be modified to incorporate a temporal dimension, represented by a discrete-time index t. Parameter updates must handle data in real time, with constraints on memory and latency [34]. One can consider an online gradient descent procedure for time series prediction. Let (x_t, y_t) be the data at time t. The parameter update after observing each sample becomes [35]

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta \mathcal{L}(f(x_t; \theta_t), y_t).$$

This system may be extended to incorporate forgetting factors or adaptive learning rates, especially in nonstationary environments. In MLaaS, these incremental algorithms are advantageous for streaming applications, but they also require robust checkpointing and fault-tolerance mechanisms to handle node failures or network disruptions.

Reinforcement learning algorithms are another frontier, where an agent learns to perform actions in an environment to maximize a cumulative reward [36]. The agent's policy $\pi(a|s)$ can be parameterized by θ, and the goal is to solve

$$\max_\theta \sum_{t=0}^{T} \mathbb{E}[\gamma^t r_t],$$

with r_t representing the reward at time t and $0 < \gamma < 1$ being a discount factor. Parallelization arises in distributing the experience collection across multiple simulators or real-world agents, then aggregating the gradients of policy networks [37]. MLaaS platforms that support reinforcement learning must address the challenges of multi-agent coordination, latency in action selection, and partial observability of the environment. Advanced distributed optimization methods, such as asynchronous advantage actor-critic approaches, highlight the need for specialized architectures and robust message-passing routines.

As models grow in complexity, computational aspects like matrix factorizations, tensor operations, and partial differential equation solvers also become relevant, especially in scientific machine learning or physics-informed neural networks [38]. Solutions to PDE-based problems, for example, may rely on neural networks to approximate the unknown function $u(\mathbf{x})$, which satisfies a governing equation

$$\mathcal{D}[u](\mathbf{x}) = 0,$$

with $\mathcal{D}$ being a differential operator. The training process enforces the residual of $\mathcal{D}[u]$ to be near zero across sampled points in the domain. Parallelizing the solution of such problems in an MLaaS context entails partitioning spatial domains and computational grids, then coordinating gradient exchanges among distributed compute nodes. This integration of computational physics and large-scale data analytics exemplifies the advanced mathematical demands placed on machine learning services.

The development of algorithmic frameworks that ensure convergence and stability under distributed operation is an active area of study [39]. The classical issues of network latencies, asynchronous updates, and fault tolerance must be rigorously incorporated into convergence analyses. For instance, system-level strategies like bounded staleness in parameter updates can be incorporated into theoretical guarantees for certain classes of convex objectives. Non-convex objectives, prevalent in deep learning, pose more complex challenges, often requiring practical heuristics to ensure that distributed training remains stable and efficient [40]. These findings feed directly into the design of resource schedulers and cluster managers, which must adapt to changing conditions in real time.

It is evident that the synergy between advanced mathematical modeling and cloud-based architectures underpins the success of MLaaS. The interplay extends beyond the mere application of well-known algorithms, requiring a rethinking of the entire lifecycle of model development, from data preprocessing to final deployment [41]. By merging high-level abstractions with optimized low-level routines, MLaaS platforms aim to hide the complexities of distribution from end users while still offering robust performance and high-quality solutions.

4. Scalability and Performance Issues

Although the conceptual and mathematical underpinnings of MLaaS promise significant benefits, real-world deployments encounter notable scalability and performance bottlenecks. These challenges often stem from the tension between high-throughput processing requirements and the inherent heterogeneity of distributed clusters [42]. Network bandwidth constraints and latency can hamper the movement of large datasets between nodes, causing training slowdowns even if substantial compute resources are available. One of the most pressing issues is the phenomenon of straggler nodes, where certain machines run more slowly than others due to hardware differences or transient failures. This heterogeneity leads to waiting times during parameter aggregation steps, reducing the overall efficiency of parallel training. [43, 44]

Elasticity is another dimension that complicates scalability. While cloud platforms are designed to add or remove resources on demand, sudden changes in cluster size can interrupt ongoing training processes if the algorithms are not equipped to adapt mid-epoch. A common approach is to checkpoint intermediate model states, allowing training to resume smoothly after nodes are scaled up or down [45]. However, frequent checkpointing can introduce overhead and increase storage costs, particularly for large models. A balanced strategy is required to manage these trade-offs. In practice, heuristics are employed to determine optimal checkpoint intervals based on metrics like training progress, cost considerations, and observed system reliability. [46]

Another focal area is data partitioning. In distributed training scenarios, data is typically split into shards, each processed by a separate subset of the cluster. Poor partitioning can lead to load imbalance, where certain nodes receive disproportionately large or computationally intensive data chunks. Monitoring data distribution in real time and redistributing shards if imbalances arise is crucial for maintaining high throughput [47]. Some MLaaS frameworks use advanced hashing techniques to ensure random but roughly uniform splits. Others adopt dynamic partitioning methods that reassign shards during runtime based on processing speed. These strategies add scheduling complexity, as they must determine whether the cost of rebalancing is justified by the performance gains. [48]

For memory-intensive workloads, techniques like model parallelism divide large neural networks across multiple nodes, effectively splitting the layers or parameters. However, this approach can result in heavy inter-node communication as the intermediate outputs of one node become the inputs for another. Minimizing these data transfers is pivotal, as communication overhead can overshadow the computational savings of parallelism [49, 50]. Optimizing data layout in memory and caching frequently accessed parameters are additional considerations for MLaaS providers aiming to maximize performance. The use of specialized high-performance interconnects in the data center, such as Remote Direct Memory Access or InfiniBand, can mitigate latency, but these solutions increase infrastructure costs and may not be universally available.

Fault tolerance is another crucial element of scalability [51]. On large clusters, node failures become statistically likely, and a single failure can jeopardize an entire training run if not handled gracefully. Checkpoint-restart strategies, redundant computations, and erasure coding for parameter storage are a few of the techniques employed to maintain progress despite hardware or software faults. Additionally, asynchronous updates can offer resilience, as the parameter server can continue to update global parameters even if a subset of worker nodes are temporarily offline [52]. This, however, brings complexities in ensuring consistent parameter versions across nodes and maintaining stable convergence.

Load balancing is essential in operational settings involving continuous data feeds and dynamic workloads. Some tasks, such as hyperparameter optimization, can be parallelized by running multiple experiments concurrently, each using different configurations [53]. The results inform a global controller that narrows down promising parameter sets. Efficient resource allocation for such multi-experiment scenarios is non-trivial, as the computational demands for each experiment vary based on model complexity and dataset size. Moreover, guaranteeing fairness among multiple tenants who share the same MLaaS infrastructure requires sophisticated scheduling policies [54, 55]. Providers must ensure that no single user monopolizes resources, while still allowing for large-scale jobs to proceed within reasonable time frames.

The increasing popularity of serverless computing has introduced further nuances in scalability. Instead of maintaining long-lived virtual machines, serverless platforms spin up ephemeral containers on demand, charging users only for the time their functions actually run [56]. Integrating machine learning workloads into such platforms reduces overhead when dealing with sporadic inference requests, but also raises complexities in cold-start latency and the management of stateful training processes. Overcoming cold-start penalties might involve keeping certain containers warm or adopting specialized container images optimized for fast boot times. Despite these challenges, serverless paradigms offer potential cost benefits and can be particularly well-suited for applications with unpredictable or bursty workloads, reinforcing the idea that MLaaS ecosystems must cater to diverse usage patterns. [57]

Additionally, emerging hardware accelerators influence scalability considerations. Graphics Processing Units, Tensor Processing Units, and Field Programmable Gate Arrays offer immense speedups for specific machine learning tasks, but each has unique programming models and performance characteristics. Integrating these accelerators into MLaaS platforms demands specialized schedulers that can match workloads to the most suitable hardware, monitor real-time performance metrics, and seamlessly switch or load-balance tasks when conditions change [58]. This tight coupling between hardware and software extends the complexity of resource orchestration, as not all workloads benefit equally from acceleration, and the actual gains often depend on data layout, batch size, and algorithmic structure.

In terms of performance metrics, throughput, latency, and cost-efficiency stand out as primary concerns. Throughput is measured in terms of how many training samples or inference requests can be processed per second, while latency highlights the response time for individual requests, which is critical for real-time or interactive applications [59]. Cost-efficiency introduces an economic dimension, compelling users to optimize resource usage relative to performance targets. This optimization necessitates advanced performance models capable of predicting how scaling decisions affect both speed and cost. Moreover, as data volumes surge, many organizations employ tiered storage, where hot data resides on expensive but fast media while cold data stays on cheaper, slower systems [60]. This hierarchical storage model adds another layer of complexity to performance tuning, since retrieving data from cold storage introduces additional latency.

Overall, the interplay between parallel computation, network communication, hardware acceleration, and fault tolerance underscores the challenges of scalability and performance in MLaaS. The intricate balancing act required to meet service-level objectives on heterogeneous, elastic infrastructures is at the heart of ongoing research in cloud-based machine learning [61]. Continuous improvements in scheduling algorithms, network technologies, and programming abstractions will likely expand the envelope of scalable performance, but some fundamental bottlenecks remain. Network bandwidth and latency, for instance, may persist as constraints even as compute power increases. As the next section will show, these technical complexities intersect significantly with security, privacy, and regulatory requirements, further complicating the design and management of MLaaS platforms. [62]

5. Security, Privacy, and Regulatory Constraints

Security and privacy are paramount considerations in Machine Learning as a Service, because data often includes sensitive customer information, proprietary corporate intelligence, or regulated content such as healthcare records and financial transactions. The growing sophistication of adversaries amplifies concerns around data breaches, model inversion attacks, and adversarial inputs that can compromise system integrity. Implementing robust defenses is a multi-layered endeavor, involving encryption at rest and in transit, secure key management, continuous monitoring, and anomaly detection systems that flag suspicious activity [63, 64]. Yet these solutions must align with performance objectives, as heavy cryptographic overhead can degrade throughput and inflate operational costs.

One important aspect lies in data governance. Large corporations and research institutions commonly handle data subject to strict regulations requiring explicit consent, data minimization, and traceability [65]. Distributing such data across multiple servers in different jurisdictions introduces legal complexities related to cross-border data flows. Mechanisms like secure enclaves and homomorphic encryption have been explored to allow computation on encrypted data without exposing raw content. However, these methods can be computationally intensive and may not be feasible for real-time processing at large scale [66]. Balancing compliance with operational viability is thus a recurrent issue in MLaaS environments.

Beyond data protection, privacy-preserving machine learning techniques play a pivotal role in mitigating risks. Differential privacy is one such approach, introducing controlled noise into the learning process to limit the leakage of individual data points [67]. Formally, a mechanism M is ε-differentially private if for any two datasets D and D' that differ by a single record, and for any output S,

$$\Pr[M(D) \in S] \leq e^{\varepsilon} \Pr[M(D') \in S].$$

Implementing differential privacy in a distributed setting requires careful coordination to ensure that each worker node adheres to privacy budgets. Additionally, advanced cryptographic protocols such as secure multiparty computation attempt to split the data among multiple participants, enabling collaborative model training without revealing individual data subsets [68]. While theoretically attractive, these approaches demand specialized algorithms to handle the overhead of secure computations, and their latency often does not meet the needs of large-scale industrial applications.

Model attacks highlight another dimension of MLaaS security. Techniques like model inversion allow attackers with access to model predictions to reconstruct sensitive features of the training data [69]. Membership inference attacks, on the other hand, can determine whether a given sample was included in the training set. MLaaS providers must integrate defenses like output obfuscation, gradient encryption, or model watermarking to detect and mitigate these threats. However, adopting such defenses might introduce accuracy losses or additional computational steps, underscoring the trade-off between security and model performance.

Adversarial machine learning poses yet another challenge, where small perturbations to input data can cause models to make significant misclassifications [70]. Implementing adversarial training, gradient masking, or robust optimization techniques can reduce vulnerability, but each countermeasure comes with computational and design complexities. MLaaS platforms operating in fields such as autonomous driving, medical diagnosis, or financial forecasting must be particularly vigilant, given the high stakes involved. Rigorous security testing, including penetration testing and red teaming, forms part of a comprehensive defense strategy.

On the regulatory side, compliance frameworks impose further constraints on how data is collected, processed, stored, and shared. Various standards mandate audit trails, data retention policies, and transparent accountability in automated decision-making processes. The concept of explainability is crucial for sensitive applications, where users or regulators might demand to know the rationale behind certain predictions [71]. While post-hoc explainability methods can be integrated to provide insights into model outputs, these frameworks can be difficult to implement when dealing with highly parallelized and containerized deployments. The ephemeral nature of containers can complicate logging, making it challenging to trace model decisions after the fact, especially if intermediate data states are not preserved.

Edge scenarios exacerbate some of these concerns [72]. When data is collected and partially processed on edge devices, encryption keys and preliminary model states must be securely managed. The connectivity between edge nodes and the cloud might be intermittent or low-bandwidth, complicating the synchronization of security policies and patch deployments. This situation can create vulnerabilities if an edge node is compromised and remains offline, unable to receive security updates or rotate cryptographic keys [73]. Additionally, the integration of personal devices, industrial sensors, or medical equipment into MLaaS pipelines amplifies the complexity of ensuring end-to-end security, given the wide range of hardware and software ecosystems involved.

Privacy regulations also demand that certain categories of data be anonymized or pseudonymized before processing. Achieving this at scale requires advanced anonymization algorithms that preserve data utility while protecting individual identities [74]. Some frameworks use k-anonymity or l-diversity measures to quantify the degree of protection, but these approaches can degrade the statistical properties of the dataset if not carefully calibrated. For MLaaS platforms that rely on high fidelity data to maintain model accuracy, finding the right balance is essential. Overly aggressive anonymization might render predictive features useless, while lenient approaches risk leaking sensitive information. [75]

Looking ahead, quantum computing and post-quantum cryptography also loom as potential disruptors. Although large-scale quantum computers have not yet been widely deployed, some experts anticipate that existing encryption algorithms may be vulnerable to future attacks by quantum adversaries. MLaaS providers with long data retention timelines may need to adopt cryptographic agility to ensure that data remains secure against quantum threats [76]. Research into quantum-safe encryption and key exchange protocols is ongoing, but wide adoption may be gradual, influenced by cost, compatibility, and regulatory inertia.

In summary, security, privacy, and regulatory constraints interweave with the technical challenges of large-scale machine learning, requiring a defense-in-depth strategy and constant vigilance. Beyond implementing cryptographic measures and secure coding practices, MLaaS platforms must invest in privacy-preserving machine learning, robust defense against adversarial attacks, and compliance with a matrix of legal frameworks [77]. This multifaceted challenge underscores the complex environment in which MLaaS operates, linking cloud infrastructure, big data, and advanced analytics under the umbrella of evolving global regulations. The final section will synthesize the findings, offer insights into future developments, and discuss realistic outcomes and limitations of current approaches.

6. Conclusion

Machine Learning as a Service has grown from a niche offering into a foundational paradigm for big data processing in the cloud [78]. By abstracting the complexities of deploying and maintaining distributed computing infrastructure, these platforms enable organizations of varying sizes to harness advanced algorithms at scale. The conceptual framework spanning data ingestion, orchestration, continuous integration, and elastic deployment has proven remarkably effective for a wide range of applications, from image recognition to financial forecasting. Simultaneously, mathematical modeling and algorithmic insights underpin the distributed nature of modern machine learning, providing guarantees and heuristics for tasks such as parameter aggregation, cluster partitioning, and online adaptation to nonstationary data streams. [79, 80]

Technical challenges in scaling remain a prominent area of focus, encompassing network latency, straggler nodes, model parallelism, and fault-tolerant training. These bottlenecks underscore the ongoing requirement for innovative scheduling algorithms, optimized communication protocols, and dynamic resource management that can handle ever-increasing data volumes and model complexities. Simultaneously, the integration of specialized hardware accelerators and emerging serverless paradigms indicates that MLaaS will continue to diversify in terms of infrastructure strategies [81]. Achieving an optimal balance between high throughput, low latency, and cost-effectiveness remains an open challenge that drives much of the current research and development efforts.

Security and privacy considerations elevate the stakes, given the sensitivity of the data and the high-profile nature of potential breaches. Defenses against data exfiltration, model inversion, and adversarial attacks must be embedded at every layer of the MLaaS stack, from encrypted data storage to privacy-preserving learning algorithms [82]. However, these solutions can impact performance and cost, pushing providers to develop nuanced trade-off analyses. Regulatory constraints add another layer of complexity, especially for multi-jurisdictional deployments that handle personal data. Frameworks for explainability, auditability, and accountability are becoming more important, particularly in sectors such as healthcare and finance, where the consequences of model misbehavior can be severe. [83]

Realistically, current MLaaS systems already demonstrate considerable success, offering high-accuracy models, user-friendly interfaces, and reliable scaling for typical enterprise workloads. Users can train large language models, image classifiers, or time-series forecasters with relative ease, paying only for the resources consumed. Yet limitations persist in scenarios that demand extremely low-latency responses under tight resource constraints, such as embedded or edge systems with poor connectivity [84]. In addition, the complexity of cloud billing models and the heterogeneity of hardware accelerators can deter users who lack specialized knowledge or the necessary budgets. Areas like reinforcement learning and PDE-based modeling, while supported in some platforms, still face performance and scalability hurdles in real-world deployments. The ongoing evolution of distributed optimization methods, combined with more mature container orchestration, is likely to address many of these issues over time. [85]

There is also a recognition that machine learning workloads can produce biased or unfair outcomes unless properly curated and audited. Although the technical community is developing methods to detect and mitigate such bias, the ethical and operational implications of deploying these solutions at scale remain significant. Moreover, the interplay between cloud providers, data owners, and end users creates a multifaceted ecosystem where trust and transparency are paramount [86]. The need for robust contracts, clear delineations of responsibility, and oversight mechanisms that transcend organizational boundaries highlights an additional dimension in which MLaaS must evolve.

As research continues, one can anticipate the emergence of hybrid architectures that blend the strengths of edge computing, on-premises clusters, and public cloud resources. Such hybrid models might employ advanced schedulers that dynamically migrate workloads among different environments, optimizing cost, latency, and compliance in real time [87]. Better integration of quantum-safe cryptographic primitives, zero-trust networking, and ephemeral container ecosystems could further strengthen the resilience and adaptability of MLaaS solutions. In terms of algorithmic advances, the rise of self-supervised and transfer learning suggests that large-scale pretraining will remain a focal point, benefiting substantially from cloud-based GPU or TPU clusters.

In conclusion, Machine Learning as a Service stands at the intersection of numerous technical, organizational, and regulatory domains [88]. It encapsulates both the promise of democratized data-driven insights and the complexities of securing, scaling, and governing large-scale analytics. The synergy between conceptual foundations, rigorous mathematical modeling, and sophisticated performance engineering underpins its rapid growth. Even as challenges persist, the continuous momentum in this field suggests that MLaaS will increasingly serve as a linchpin for next-generation innovations across industries. Future progress in addressing open issues around scalability, security, and compliance will likely cement its role as a pivotal enabler of real-time, data-centric decision making. [89]

References

[1] S. Witvoet, D. de Massari, S. Shi, and A. F. Chen, "Leveraging large, real-world data through machine-learning to increase efficiency in robotic-assisted total knee arthroplasty.," *Knee surgery, sports traumatology, arthroscopy : official journal of the ESSKA*, vol. 31, pp. 3160–3171, 1 2023.

[2] I. Lee, "An optimization approach to capacity evaluation and investment decision of hybrid cloud: a corporate customer's perspective," *Journal of Cloud Computing*, vol. 8, pp. 1–13, 11 2019.

[3] O. Loyola-González, M. A. Medina-Pérez, and K.-K. R. Choo, "A review of supervised classification based on contrast patterns: Applications, trends, and challenges," *Journal of grid computing*, vol. 18, pp. 797–845, 10 2020.

[4] A. L. S. Saabith, E. A. Sundararajan, and A. A. Bakar, "A parallel apriori-transaction reduction algorithm using hadoop-mapreduce in cloud," *Asian Journal of Research in Computer Science*, pp. 1–24, 4 2018.

[5] Y. Asaki, B. A. Pampliega, P. G. Edwards, S. Iguchi, and E. J. Murphy, "Astronomical radio interferometry," *Nature Reviews Methods Primers*, vol. 3, 11 2023.

[6] G. A. Gravvanis, J. P. Morrison, D. C. Marinescu, and C. K. Filelis-Papadopoulos, "Special section: towards high performance computing in the cloud," *The Journal of Supercomputing*, vol. 74, pp. 527–529, 1 2018.

[7] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, “Federated query processing for big data in data science,” in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 6145–6147, IEEE, 2019.

[8] S. Mangul, L. S. Martin, B. Langmead, J. E. Sanchez-Galan, I. Toma, F. Hormozdiari, P. A. Pevzner, and E. Eskin, “How bioinformatics and open data can boost basic science in countries and universities with limited resources.,” *Nature biotechnology*, vol. 37, pp. 324–326, 3 2019.

[9] N. Matsushima, H. Miyashita, and R. H. Kretsinger, “Sequence features, structure, ligand interaction, and diseases in small leucine rich repeat proteoglycans.,” *Journal of cell communication and signaling*, vol. 15, pp. 1–13, 4 2021.

[10] X. Sun and N. Ansari, “Adaptive avatar handoff in the cloudlet network,” *IEEE Transactions on Cloud Computing*, vol. 7, pp. 664–676, 7 2019.

[11] V. Navale and P. E. Bourne, “Cloud computing applications for biomedical science: A perspective,” *PLoS computational biology*, vol. 14, pp. e1006144–, 6 2018.

[12] A. Malik, S. Gupta, and S. Dhall, “Analysis of traditional and modern image encryption algorithms under realistic ambience,” *Multimedia Tools and Applications*, vol. 79, pp. 27941–27993, 7 2020.

[13] X. Xu, “To social with social distance: a case study on a vr-enabled graduation celebration amidst the pandemic.,” *Virtual reality*, vol. 27, pp. 1–3331, 4 2022.

[14] T. Riccardi, L. Malatesta, K. V. Damme, A. S. Suleiman, A. Farcomeni, M. Rezende, P. Vahalík, and F. Attorre, “Environmental factors and human activity as drivers of tree cover and density on the island of socotra, yemen,” *Rendiconti Lincei. Scienze Fisiche e Naturali*, vol. 31, pp. 703–718, 6 2020.

[15] M. Abouelyazid and C. Xiang, “Architectures for ai integration in next-generation cloud infrastructure, development, security, and management,” *International Journal of Information and Cybersecurity*, vol. 3, no. 1, pp. 1–19, 2019.

[16] null Odunayo Josephine Akindote, null Abimbola Oluwatoyin Adegbite, null Samuel Onimisi Dawodu, null Adedolapo Omotosho, and null Anthony Anyanwu, “Innovation in data storage technologies: From cloud computing to edge computing,” *Computer Science & IT Research Journal*, vol. 4, pp. 273–299, 12 2023.

[17] M. Vita, K. Brand, M. Larson-Koester, N. Petek, C. Taragin, W. Violette, and D. H. Wood, “Economics at the ftc: Estimating harm from deception and analyzing mergers,” *Review of Industrial Organization*, vol. 61, pp. 405–438, 11 2022.

[18] M. Nadin and A. Naz, “Architecture as service: a case of design on demand (dod),” *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, pp. 4751–4769, 12 2018.

[19] R. Gratton, A. Bragaglia, E. Carretta, V. D’Orazi, S. Lucatello, and A. Sollima, “What is a globular cluster? an observational perspective,” *The Astronomy and Astrophysics Review*, vol. 27, pp. 1–136, 11 2019.

[20] R. Avula, “Architectural frameworks for big data analytics in patient-centric healthcare systems: Opportunities, challenges, and limitations,” *Emerging Trends in Machine Intelligence and Big Data*, vol. 10, no. 3, pp. 13–27, 2018.

[21] K. Miyasaka, K. H. Shelley, S. Takahashi, H. Kubota, K. Ito, I. Yoshiya, A. Yamanishi, J. B. Cooper, D. J. Steward, H. Nishida, J. Kiani, H. Ogino, Y. Sata, R. J. Kopotic, K. Jenkin, A. Hannenberg, and A. A. Gawande, “Tribute to dr. takuo aoyagi, inventor of pulse oximetry.,” *Journal of anesthesia*, vol. 35, pp. 671–709, 8 2021.

[22] A. Curioni, “Artificial intelligence: Why we must get it right,” *Informatik-Spektrum*, vol. 41, pp. 7–14, 2 2018.

[23] Y. Luo, “A general framework of digitization risks in international business.,” *Journal of international business studies*, vol. 53, pp. 1–18, 5 2021.

[24] P. Schwerdtfeger, O. R. Smits, and P. Pyykkö, “The periodic table and the physics that drives it,” *Nature reviews. Chemistry*, vol. 4, pp. 359–380, 6 2020.

[25] R. Z. Naeem, S. Bashir, M. F. Amjad, H. Abbas, and H. Afzal, “Fog computing in internet of things: Practical applications and future directions,” *Peer-to-Peer Networking and Applications*, vol. 12, pp. 1236–1262, 3 2019.

[26] A. K. Saxena, “Evaluating the regulatory and policy recommendations for promoting information diversity in the digital age,” *International Journal of Responsible Artificial Intelligence*, vol. 11, no. 8, pp. 33–42, 2021.

[27] F. L. Bookstein, “Reflections on a biometrics of organismal form,” *Biological Theory*, vol. 14, pp. 177–211, 4 2019.

[28] C. G. Rodrigues and M. Stanley, “Cad software and its influence on complex treatment planning,” *Current Oral Health Reports*, vol. 10, pp. 59–68, 6 2023.

[29] N. Baydeti, R. Veilumuthu, and M. Vaithilingam, “Scalable models for redundant data flow analysis in online social networks,” *Wireless Personal Communications*, vol. 107, pp. 2123–2142, 4 2019.

[30] S. Jauhar, S. Pratap, null Lakshay, S. Paul, and A. Gunasekaran, “Internet of things based innovative solutions and emerging research clusters in circular economy,” *Operations Management Research*, vol. 16, pp. 1968–1988, 10 2023.

[31] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, “Approximate query processing for big data in heterogeneous databases,” in *2020 IEEE international conference on big data (big data)*, pp. 5765–5767, IEEE, 2020.

[32] N. Maleki, H. R. Faragardi, A. M. Rahmani, M. Conti, and J. Lofstead, “Tmar: a two-stage mapreduce scheduler for heterogeneous environments,” *Human-centric Computing and Information Sciences*, vol. 10, pp. 1–26, 10 2020.

[33] M. Khan, M. A. A. Irfan, and N. Ullah, “Measurement of traffic-related air pollution in peshawar, pakistan - a pilot study,” *Asian Journal of Atmospheric Environment*, vol. 15, 12 2021.

[34] D. H. Shugar, A. Burr, U. K. Haritashya, J. S. Kargel, C. S. Watson, M. C. Kennedy, A. R. Bevington, R. Betts, S. Harrison, and K. Strattman, “Rapid worldwide growth of glacial lakes since 1990,” *Nature Climate Change*, vol. 10, pp. 939–945, 8 2020.

[35] R. K. Barik, C. Misra, R. K. Lenka, H. Dubey, and K. Mankodiya, “Hybrid mist-cloud systems for large scale geospatial big data analytics and processing: opportunities and challenges,” *Arabian Journal of Geosciences*, vol. 12, pp. 1–15, 1 2019.

[36] C.-T. Yang, J.-C. Liu, Y.-W. Chan, E. Kristiani, and C.-F. Kuo, “Performance benchmarking of deep learning framework on intel xeon phi,” *The Journal of Supercomputing*, vol. 77, pp. 2486–2510, 6 2020.

[37] D. Nathan and N. Ahmed, “Technological change and employment: Creative destruction,” *The Indian Journal of Labour Economics*, vol. 61, pp. 281–298, 10 2018.

[38] B. Karmakar, S. Das, S. Bhattacharya, R. Sarkar, and I. Mukhopadhyay, “Tight clustering for large datasets with an application to gene expression data,” *Scientific reports*, vol. 9, pp. 3053–, 2 2019.

[39] A. Gregg, J. Yu, J. Resig, L. Johnson, E. Park, and P. Stuczynski, “Promising educational technology meets complex system: a 6-year case study of an adaptive learning project from initial exploration through the end of a pilot,” *Journal of Formative Design in Learning*, vol. 5, pp. 62–77, 7 2021.

[40] N. Charon, B. Charlier, and A. Trouvé, “Metamorphoses of functional shapes in sobolev spaces,” *Foundations of Computational Mathematics*, vol. 18, pp. 1535–1596, 1 2018.

[41] R. Ahmad, I. Alsmadi, and M. Al-Ramahi, “Optimization of deep learning models: benchmark and analysis,” *Advances in Computational Intelligence*, vol. 3, 3 2023.

[42] S. C. Tan, C. K. K. Chan, K. Bielaczyc, L. Ma, M. Scardamalia, and C. Bereiter, “Knowledge building: Aligning education with needs for knowledge creation in the digital age.,” *Educational Technology Research and Development*, vol. 69, pp. 2243–2266, 1 2021.

[43] S. L. Ustin and E. M. Middleton, “Current and near-term advances in earth observation for ecological applications,” *Ecological processes*, vol. 10, pp. 1–57, 1 2021.

[44] R. Avula, “Overcoming data silos in healthcare with strategies for enhancing integration and interoperability to improve clinical and operational efficiency,” *Journal of Advanced Analytics in Healthcare Management*, vol. 4, no. 10, pp. 26–44, 2020.

[45] X. Du and E. B. Meier, “Innovating pedagogical practices through professional development in computer science education,” *Journal of Computer Science Research*, vol. 5, pp. 46–56, 7 2023.

[46] P. Kijsanayothin, G. Chalumporn, and R. Hewett, “On using mapreduce to scale algorithms for big data analytics: a case study,” *Journal of Big Data*, vol. 6, pp. 1–20, 11 2019.

[47] K. Gairaa, S. Benkaciali, and M. Guermoui, “Clear-sky models evaluation of two sites over algeria for pv forecasting purpose,” *The European Physical Journal Plus*, vol. 134, pp. 534–, 10 2019.

[48] G. Zhang, J. Gao, and M. Du, “Parameterized forward operators for simulation and assimilation of polarimetric radar data with numerical weather predictions,” *Advances in Atmospheric Sciences*, vol. 38, pp. 737–754, 4 2021.

[49] J. Scheibmeir and Y. K. Malaiya, "Social media analytics of the internet of things," *Discover Internet of Things*, vol. 1, pp. 1–15, 7 2021.

[50] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, "Automatic visual recommendation for data science and analytics," in *Advances in Information and Communication: Proceedings of the 2020 Future of Information and Communication Conference (FICC), Volume 2*, pp. 125–132, Springer, 2020.

[51] R. Z. Yousif, S. W. Kareem, and S. M. J. Abdalwahid, "Enhancing approach for information security in hadoop," *Polytechnic Journal*, vol. 10, pp. 81–87, 6 2020.

[52] Q. M. Le, A. Amer, and J. Holliday, "Raid 4smr: Raid array with shingled magnetic recording disk for mass storage systems," *Journal of Computer Science and Technology*, vol. 34, pp. 854–868, 7 2019.

[53] Z. Shi and G. Wang, "Integration of big-data erp and business analytics (ba)," *The Journal of High Technology Management Research*, vol. 29, no. 2, pp. 141–150, 2018.

[54] S. Ramlo, "The coronavirus and higher education: Faculty viewpoints about universities moving online during a worldwide pandemic.," *Innovative higher education*, vol. 46, pp. 1–19, 1 2021.

[55] M. Kansara, "A framework for automation of cloud migrations for efficiency, scalability, and robust security across diverse infrastructures," *Quarterly Journal of Emerging Technologies and Innovations*, vol. 8, no. 2, pp. 173–189, 2023.

[56] A. Alsadi, L. Fu, N. K. Majeed, Y. Dharmamer, U. Edomwonyi, C. Baste, J. V. Groth, M. Singh, T. Patel, K. Dowlatshahi, A. Pintado, C. Geary, J. C. McClay, J. R. Campbell, W. S. Campbell, J. Hudeček, L. Voorwerk, J. van den Berg, K. V. de Vijver, M. Kok, R. Salgado, H. M. Horlings, Y. Ishida, M. Tsuda, J. Suzuka, L. Wang, S. Tanikawa, H. Sugino, S. Tanaka, D. B. Joseph, S. Arvisais-Anhalt, H. Hwang, Y. Fang, Y. Peng, K. Gwin, V. Sarode, S. Sahoo, E. Araj, Y. Kim, and M. H. A. Roehrl, "Abstracts from uscap 2019: Informatics (1463-1508).," *Modern pathology : an official journal of the United States and Canadian Academy of Pathology, Inc*, vol. 32, no. Suppl 2, pp. 1–40, 2019.

[57] M. Padovani, A. V. Ivlev, D. Galli, S. S. R. Offner, N. Indriolo, D. Rodgers-Lee, A. Marcowith, P. Girichidis, A. M. Bykov, and J. M. D. Kruijssen, "Impact of low-energy cosmic rays on star formation," *Space Science Reviews*, vol. 216, pp. 29–, 3 2020.

[58] H. G. Amlak, A. M. Jabbari, Y. Chen, B. Y. Choi, C. T. Huang, and S. Song, "Agile polymorphic software-defined fog computing platform for mobile wireless controllers and sensors," *International Journal of Internet Technology and Secured Transactions*, vol. 9, no. 4, pp. 426–426, 2019.

[59] L. Luciano and A. B. Hamza, "Deep similarity network fusion for 3d shape classification," *The Visual Computer*, vol. 35, pp. 1171–1180, 5 2019.

[60] S. K. S. Cheung, L. F. Kwok, K. Phusavat, and H. H. Yang, "Shaping the future learning environments with smart elements: challenges and opportunities.," *International journal of educational technology in higher education*, vol. 18, pp. 1–9, 3 2021.

[61] W. Tong, L. Li, X. Zhou, and J. Franklin, "Efficient spatiotemporal interpolation with spark machine learning," *Earth Science Informatics*, vol. 12, pp. 87–96, 11 2018.

[62] J. Chen and H. Wang, "Guest editorial: Big data infrastructure ii," *IEEE Transactions on Big Data*, vol. 4, pp. 299–300, 9 2018.

[63] N.-E. Omrani, F. Ogawa, H. Nakamura, N. Keenlyside, S. W. Lubis, and K. Matthes, "Key role of the ocean western boundary currents in shaping the northern hemisphere climate," *Scientific reports*, vol. 9, pp. 3014–, 2 2019.

[64] R. Avula, "Assessing the impact of data quality on predictive analytics in healthcare: Strategies, tools, and techniques for ensuring accuracy, completeness, and timeliness in electronic health records," *Sage Science Review of Applied Machine Learning*, vol. 4, no. 2, pp. 31–47, 2021.

[65] A. Al-Sinayyid and M. Zhu, "Job scheduler for streaming applications in heterogeneous distributed processing systems," *The Journal of Supercomputing*, vol. 76, pp. 9609–9628, 3 2020.

[66] A.-U.-H. Yasar, H. Malik, and E. M. Shakshuki, "Special issue on ubiquitous computing and nextgen context-fusion," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, pp. 4625–4627, 5 2018.

[67] S. Kadioglu, "Core group placement: allocation and provisioning of heterogeneous resources," *EURO Journal on Computational Optimization*, vol. 7, no. 3, pp. 243–264, 2019.

[68] S. Chan, V. Reddy, B. Myers, Q. Thibodeaux, N. Brownstone, and W. Liao, “Machine learning in dermatology: Current applications, opportunities, and limitations,” *Dermatology and therapy*, vol. 10, pp. 365–386, 4 2020.

[69] F. Salahdine, T. Han, and N. Zhang, “5g, 6g, and beyond: Recent advances and future challenges,” *Annals of Telecommunications*, vol. 78, pp. 525–549, 1 2023.

[70] K. Katsaliaki, P. Galetsi, and S. Kumar, “Supply chain disruptions and resilience: a major review and future research agenda,” *Annals of operations research*, vol. 319, pp. 1–38, 1 2021.

[71] R. Dillerup, T. Witzemann, and B. Schröckhaas, “Zehn trends der unternehmensplanung,” *Controlling & Management Review*, vol. 64, pp. 46–54, 4 2020.

[72] H. Kuwajima, H. Yasuoka, and T. Nakae, “Engineering problems in machine learning systems,” *Machine Learning*, vol. 109, pp. 1103–1126, 4 2020.

[73] F. Firouzi, B. Farahani, and A. Marinsek, “The convergence and interplay of edge, fog, and cloud in the ai-driven internet of things (iot),” *Information Systems*, vol. 107, pp. 101840–, 2022.

[74] M. Servi, F. Buonamici, R. Furferi, L. Governi, F. Uccheddu, Y. Volpe, S. Leng, F. Facchini, M. Ghionzoli, and A. Messineo, “Pectus carinatum: a non-invasive and objective measurement of severity.,” *Medical & biological engineering & computing*, vol. 57, pp. 1727–1735, 6 2019.

[75] R. J. Longman, T. W. Giambelluca, M. A. Nullet, A. G. Frazier, K. Kodama, S. D. Crausbay, P. D. Krushelnycky, S. Cordell, M. P. Clark, A. J. Newman, and J. R. Arnold, “Compilation of climate data from heterogeneous networks across the hawaiian islands.,” *Scientific data*, vol. 5, pp. 180012–180012, 2 2018.

[76] F. Spahn, M. Sachse, M. Seiß, H.-W. Hsu, S. Kempf, and M. Horanyi, “Circumplanetary dust populations,” *Space Science Reviews*, vol. 215, pp. 1–54, 1 2019.

[77] J. Halverson, B. D. Nelson, and F. Ruehle, “Branes with brains: Exploring string vacua with deep reinforcement learning,” *Journal of High Energy Physics*, vol. 2019, pp. 003–, 6 2019.

[78] M. N. İNCE, M. GÜNAY, and J. LEDET, “Lightweight distributed computing framework for orchestrating high performance computing and big data,” *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 30, pp. 1571–1585, 5 2022.

[79] D. Gorman and T. M. Kashner, “Medical graduates, truthful and useful analytics with big data, and the art of persuasion.,” *Academic medicine : journal of the Association of American Medical Colleges*, vol. 93, no. 8, pp. 1113–1116, 2018.

[80] M. Abouelyazid, “Forecasting resource usage in cloud environments using temporal convolutional networks,” *Applied Research in Artificial Intelligence and Cloud Computing*, vol. 5, no. 1, pp. 179–194, 2022.

[81] J. Xu and B. Palanisamy, “Optimized contract-based model for resource allocation in federated geo-distributed clouds,” *IEEE Transactions on Services Computing*, vol. 14, pp. 530–543, 3 2021.

[82] A. P. Plageras, K. E. Psannis, C. Stergiou, H. Wang, and B. B. Gupta, “Efficient iot-based sensor big data collection–processing and analysis in smart buildings,” *Future Generation Computer Systems*, vol. 82, pp. 349–357, 2018.

[83] M. B. Imerman and F. J. Fabozzi, “Cashing in on innovation: a taxonomy of fintech,” *Journal of Asset Management*, vol. 21, pp. 167–177, 5 2020.

[84] R. K. Barik, R. Priyadarshini, R. K. Lenka, H. Dubey, and K. Mankodiya, “Fog computing architecture for scalable processing of geospatial big data,” *International Journal of Applied Geospatial Research*, vol. 11, no. 1, pp. 1–20, 2020.

[85] C. Hogendorn and B. M. Frischmann, “Infrastructure and general purpose technologies: a technology flow framework,” *European Journal of Law and Economics*, vol. 50, pp. 469–488, 2 2020.

[86] T. A. Hall and S. Hasan, “Organizational decision-making and the returns to experimentation,” *Journal of Organization Design*, vol. 11, pp. 129–144, 1 2023.

[87] A. N. Richter and T. M. Khoshgoftaar, “Melanoma risk modeling from limited positive samples,” *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 8, pp. 1–9, 4 2019.

[88] S. Derrible, “An approach to designing sustainable urban infrastructure,” *MRS Energy & Sustainability*, vol. 5, pp. 1–15, 10 2018.

[89] A. Mohammadzadeh, M. A. Zarkesh, P. H. Shahmohamd, J. Akhavan, and A. Chhabra, “Energy-aware workflow scheduling in fog computing using a hybrid chaotic algorithm,” *The Journal of Supercomputing*, vol. 79, pp. 18569–18604, 5 2023.