Energy Consumption and Optimization Strategies of Cloud-Based Big Data and Machine Learning Applications: Current Trends and Future Directions

Zhuoran Song
Vrije Universiteit Amsterdam
The Netherlands
z.song2@student.vu.nl

Matthijs Jansen
Vrije Universiteit Amsterdam
The Netherlands
m.s.jansen@vu.nl

Daniele Bonetta
Vrije Universiteit Amsterdam
The Netherlands
d.bonetta@vu.nl

Abstract

The rapid expansion of cloud computing has transformed big data analytics and machine learning, unlocking new levels of scalability and innovation. However, this progress comes with a significant environmental price—cloud data centres now account for 1-2% of global electricity consumption [7], a figure expected to rise as computational workloads grow. The environmental toll of this energy demand and inefficiencies in operations highlight the urgent need for sustainable strategies. This survey draws from cloud-based big data and ML research to provide a detailed look at energy consumption trends, assess optimization methods, and explore new technologies to enhance energy efficiency, it also delves into key areas like distributed storage systems, task scheduling, model training, and hardware utilization, weighing the trade-offs between performance and energy costs. One of the key takeaways is that a multi-layered approach—combining algorithmic improvements, infrastructure upgrades, and the integration of renewable energy—is crucial for balancing computational needs with environmental responsibility. By examining current trends and future possibilities, this study emphasizes collaborative innovation's importance in creating energy-efficient cloud ecosystems that support global sustainability goals.

1 Introduction

Cloud computing has become essential for many applications due to its scalability, flexibility, and cost-effectiveness. Big data and machine learning are two of the highly impactful fields. Big data analytics, capable of processing and analyzing vast amounts of data, and machine learning, which enables systems to learn from data and make predictions, are reshaping industries and driving innovation at an unprecedented speed [64]. However, as these technologies continue to scale and become more computationally intensive, they also intensify concerns regarding energy consumption. Cloud data centres, the backbone of these applications, consume an ever-increasing amount of energy. This ever-increasing energy consumption of cloud data centres increases operational costs. It has significant environmental implications, as the energy demands of data centres account for a substantial portion of global electricity consumption [51].

In recent years, cloud computing has experienced exponential growth. According to a report by MarketsandMarkets, the global cloud computing market size is expected to reach a particular value by a specific year, growing at a compound annual growth rate (CAGR) of 15.1% [29]. The increasing adoption of cloud-based services across various industries, including finance, healthcare, and e-commerce, drives this growth. In today's economy, data has emerged as a resource surpassing traditional commodities in value—often termed "the new oil" or "digital gold"—with organizations leveraging it to drive innovation, optimize operations, and gain competitive advantages [16].Big data and machine learning have become integral to unlocking this value, transforming raw data into actionable insights. Big data analytics (BDA) is often regarded as the intersection of machine learning (ML) and cloud computing (CC) [59], enabling large-scale data processing and enhancing analytical precision. For instance, industries such as transportation and manufacturing now rely on AI-driven data platforms to predict demand, reduce costs, and improve sustainability, with sectors like smart transportation utilizing machine learning for real-time decision-making and resource optimization [52]. Globally, the digital economy—fueled by data-centric technologies—has grown at an annual rate of 16.6%, with core digital industries contributing 7.8% to China's GDP alone [16]. Cloud computing provides the necessary infrastructure for these applications to prosper, offering scalable storage and high-performance computing resources that democratize access to advanced analytics for enterprises of all sizes.

In cloud environments, big data and machine learning fields face several high energy consumption issues, which can be broken down into several sub-problems: In big data analytics, data storage and transmission are energy-intensive. Distributed storage systems, although offering scalability and

fault tolerance, consume a significant amount of energy due to data replication and complex I/O operations [1]. Data transmission between users and cloud servers also incurs substantial energy costs, especially for remote data access and applications with high-latency connections. In machine learning, model training is highly energy-consuming. Deep learning models, such as convolutional neural networks (CNNs) and large-scale language models like GPT-3, require massive computational resources for training, leading to high energy demands [41]. Additionally, the inference phase has energy-related challenges, with factors like batch size and model architecture influencing energy consumption. At the system level, resource scheduling and management in cloud data centres are crucial for energy efficiency. Traditional scheduling algorithms may not be optimized for energy consumption, and the dynamic nature of big data and machine learning workloads makes it challenging to allocate resources efficiently. Moreover, integrating multiple optimization techniques in cloud systems adds complexity to system management, impacting energy efficiency.

Therefore, researchers have implemented optimization strategies across various dimensions, such as algorithm adjustments, resource management, and hardware utilization, in an attempt to address the energy consumption issues in big data and machine learning applications within cloud environments. For example, adaptive query processing techniques in big data analytics reduce data movement and energy consumption during query execution [2]. In machine learning, model compression techniques like pruning and knowledge distillation are applied to reduce model size and energy consumption. Leveraging specialized hardware, such as GPUs and TPUs, which offer higher computational efficiency per watt compared to traditional CPUs, and combining this with software-based optimizations, like efficient workload distribution across multiple GPUs or TPUs, can significantly improve energy efficiency [4]. Adopting energy-aware architectures in cloud data centres, which involve integrating renewable energy sources, implementing advanced cooling systems, and using energy-monitoring tools to optimize system configurations, is also crucial.

To gain a deeper understanding of the energy consumption characteristics, optimization strategies, and recent developments in the fields of big data analysis and machine learning within cloud environments, this article proposes the following three research questions:

RQ1: What are the current energy consumption characteristics of big data and machine learning applications in cloud environments?

RQ2: What are the main optimization strategies currently employed in these two fields, and what are their limitations?

RQ3: What are the future technologies and directions for energy-efficient cloud computing in big data and machine learning?

This survey offers several valuable contributions. Chapter 2 provides a detailed explanation of why big data and machine learning are chosen as the focal points of this literature review, emphasizing their significance, widespread application, and the urgent need for investigations into their energy consumption. Chapter 3 outlines the overall structure and research methodology of this paper. In chapter 4, it provides an in-depth analysis of the energy consumption patterns in big data and machine learning applications within cloud environments, covering multiple stages—such as data storage, model training, inference, and data transmission—and identifies the key factors driving energy usage at each step. In chapter 5-9, it also reviews a broad spectrum of existing optimization techniques, highlighting their strengths and limitations. These strategies range from improving distributed storage systems and data partitioning/replication to model miniaturization and optimizing hardware utilization in machine learning systems. By understanding these approaches and their constraints, researchers and practitioners are better equipped to make informed decisions about energy-saving initiatives.

Additionally, from chapter 10-11, the paper delves into emerging technologies and future strategies for boosting energy efficiency. These include advancements in physical infrastructure, green energy integration, intelligent resource management, data-centric optimizations, and cloud-native big data analytics and machine learning architectures. These insights guide future research and development toward more sustainable cloud computing solutions.

2 Why Big Data and Machine Learning?

Optimizing energy consumption is a growing challenge as cloud computing becomes a cornerstone of modern IT systems. Addressing this issue requires a thorough analysis of energy usage patterns and the development of practical optimization strategies. Big data and machine learning, as influential technologies within cloud computing, demand substantial energy and offer opportunities to improve efficiency. Focusing on these fields provides valuable insights into their energy dynamics and potential solutions.

2.1 Big Data and Machine Learning Are Core Application Domains in Cloud Computing

Cloud computing has become the backbone of modern IT systems, enabling the scalable and elastic infrastructure required for critical technologies like big data analytics (BDA) and machine learning (ML) [59]. Wu et al. emphasize that integrating ML and BDA within cloud environments is transformative; cloud computing provides ample resources for handling large-scale data, like PB - EB datasets with Hadoop's technologies, accelerating processing as seen in some famous

cases [59]. ML uncovers insights into BDA, revolutionizing business decision-making by predicting customer demands. The cloud's elasticity allows for resource adjustment and cutting costs; for example, the on-demand instances provided by Amazon Web Service (AWS) allow users to flexibly select computing resources based on actual needs, saving costs while meeting business needs, breaking the limitations of resource allocation in traditional data processing environments. Also, this integration has led to the development of innovative platforms such as Spark and Flink. Spark is faster for some workloads, and Flink can handle real-time data well, promoting big-data field progress [59]. This synergy positions ML and BDA as core drivers of innovation in cloud computing. A research shows that intense reinforcement learning can determine dynamic resource allocation according to the system's current state [64]. This maximizes the utilization efficiency of cloud resources and reduces user waiting time. BDA can analyze vast amounts of cloud data to gain insights for better resource management decisions. Their computational demands directly rely on the cloud's ability to provide on-demand resources. With the cloud's support, ML and BDA can process data more effectively, optimize resource scheduling, and drive continuous innovation in cloud computing [64].

The interconnection between big data and machine learning is clear when considering their shared technical requirements. Big data applications rely on cloud-based distributed systems to manage vast datasets [59], while machine learning models depend on cloud resources for both training and deployment [45]. As Soni and Kumar highlight, "Machine learning techniques are essential for optimizing cloud resource utilization and ensuring Quality of Service" [45]. This mutual dependence underscores the importance of studying big data analytics and machine learning within cloud environments. Their co-evolution not only drives progress but is key to shaping the future of energy-efficient and intelligent computing systems.

Integrating ML, BDA, and cloud computing creates a feedback loop: cloud platforms enable advanced analytics, while ML-driven optimization improves cloud efficiency. Historically, adopting ML on cloud infrastructure has proven to be the most cost-effective approach for BDA [59], further solidifying their combined significance.

Observation (O-1): Big data analytics (BDA) and machine learning (ML) in cloud computing environments are interdependent. BDA provides the foundational data for ML models, while ML enhances the optimization and insights derived from data analysis. This relationship fosters the simultaneous advancement of both fields, particularly in resource utilization, as cloud platforms offer scalable resources that drive innovation and performance improvements in both domains.

2.2 Energy Optimization Is Especially Crucial in Big Data and Machine Learning

However, as these technologies advance and scale, they also introduce significant energy consumption challenges. The immense computational power required for large-scale data processing and model training translates to substantial energy costs. Facebook processes over 500 terabytes of new data daily, necessitating significant computational resources and energy [14]. Similarly, the computational power needed for machine learning and intense learning models presents unique energy challenges.

As Garcia-Martín et al. note, "Machine learning researchers have primarily focused on creating highly accurate models without prioritizing energy consumption." [15] This emphasis on accuracy has led to the development of increasingly complex models with high computational and memory demands, often requiring processing power in the GigaFlops[1] range. In machine learning, models with high computational needs could perform many calculations, especially during training [15].Moreover, these models typically involve millions of parameters—values the model learns during training. For instance, the weights between neurons are considered parameters in a neural network. While many parameters give the model more flexibility to fit complex patterns in the data, it also means more significant memory usage and more computational power are needed to update them during training [35].

These energy demands do not stop once training is complete. The inference phase, where the model is deployed to make predictions or decisions, also requires significant computational power. As Garcia-Martín et al. further observes, these models may be repeatedly deployed during inference, exacerbating the need for energy-efficient solutions, particularly in real-world applications like real-time data processing and mobile platforms [15].

The trend toward more energy-intensive models is particularly evident in AI applications. The energy requirements for AI training have surged in recent years. For example, training GPT-3, a large-scale language model, consumed an estimated 1287 megawatt hours, equivalent to the annual energy usage of 120 US households [41]. This highlights the immense energy footprint of modern AI models, underscoring the urgency of developing more energy-efficient algorithms and infrastructures.

This energy-intensive trajectory is compounded by the role of cloud computing data centres, which often host these machine learning workloads. Such centres account for a significant share of global energy consumption, estimated at 1-2% of total electricity use, with projections indicating further growth in the coming years [51]. The need for effective energy optimization in cloud data centres is thus especially pronounced when dealing with the computational demands of big data and

[1] Billions of floating-point operations per second, to measure a computer's ability to process data

machine learning applications.

2.3 The Optimization Strategies in Both Domains Are Broad and Representative

The optimization techniques in big data and machine learning are broad and highly adaptable, designed to tackle various industry challenges. These approaches are as diverse as they are effective, keeping pace with rapid technological advancements and the increasing complexity of data processing and model training.

Data optimization, particularly in big data, requires diverse strategies tailored to the unique characteristics of different data types and problem domains [65]. Zhou et al. highlight that big data allows for parallel learning from multiple perspectives and granularities, which leads to the development of optimization techniques suited to various analytical tasks [65]. These strategies include distributed feature selection, adaptive scaling for high-dimensional datasets, and frameworks based on spectral graph theory for supervised and unsupervised learning. The versatility of these methods across different data contexts underscores the widespread applicability of optimization, proving its ability to handle not just traditional data processing but also more specialized analytical challenges [65].

Similarly, optimization strategies in machine learning are vast and varied, reflecting the complexity of the problems they aim to solve—for example, profound learning benefits from various techniques designed to enhance training processes and improve model performance [65]. Zhou et al. point out that deep learning can recognize a wider array of categories, surpassing traditional neural networks by multiple orders of magnitude [65]. Techniques like stochastic gradient descent, adaptive learning rates, and regularization are critical in improving accuracy while reducing computational demands, these optimization methods are key to making machine learning models more energy-efficient, showcasing their essential role in improving both the scalability and efficiency of machine learning applications [65].

These optimization efforts extend beyond the algorithmic level to the infrastructure of cloud environments. In cloud systems, where the demand for computational resources is exceptionally high, the scope of optimization strategies becomes even more significant. The Energy-Efficient Hybrid (EEH) framework, introduced by Abd El-Samie et al., combines software-level scheduling with hardware consolidation to reduce energy consumption in cloud data centres [3]. By optimizing the hardware and software layers, the EEH framework demonstrates how these strategies can effectively manage the energy needs of large-scale cloud systems without compromising performance.

Additionally, the optimization techniques used in big data and machine learning are not confined to their respective domains. Both fields benefit from the cross-pollination of parallel and distributed computing strategies [65]. These approaches, which enable the efficient processing of massive datasets and parallel model training, are crucial for the energy-efficient operation of big data analytics and machine learning systems. Zhou et al. emphasize that big data provides valuable opportunities for causal inference, which aids in decision-making [65]. In parallel, machine learning takes advantage of data and model parallelism to efficiently train models across distributed systems, reinforcing the case for optimization strategies that bridge these two fields.

This cross-domain synergy is also evident in the expanding application of big data optimization techniques outside traditional data analytics. Methods like dimensionality reduction and efficient sampling are now applied in bioinformatics, social network analysis, and other data-heavy fields [65]. Likewise, machine learning optimization techniques, including gradient-based optimization and hyperparameter tuning, are finding applications across disciplines like engineering, physics, and scientific computing, demonstrating their broad and representative impact [65].

3 Research Methodology

To systematically identify the energy consumption characteristics of big data and machine learning applications in cloud environments, this chapter adopts a structured literature review approach. The methodology consists of the following steps:

Literature Search: We conducted a comprehensive search on Google Scholar, IEEE Xplore, and ACM Digital Library using keyword combinations such as "energy consumption big data cloud," "machine learning energy optimization," and "cloud storage energy dynamics." The search focused on publications from recent years to ensure relevance to current trends.

Inclusion Criteria: Studies were included if they (1) explicitly discussed quantitative metrics for energy consumption, (2) focused on cloud computing, big data, or machine learning, and (3) proposed optimization strategies. Non-empirical studies and non-English publications were excluded.

Data Extraction: From the final set of 67 selected papers, we extracted the following information:

Key energy consumption phases (e.g., data storage, model training).

Optimization strategies and limitations (e.g., data Partitioning and replication, model miniaturization).

Future directions (e.g., AI-driven energy management, green energy integration).

3.1 Classification Framework

In this chapter, we introduce a comprehensive classification framework designed to analyze energy consumption patterns and optimization strategies within big data analytics (BDA) and machine learning (ML). The framework is built around several key dimensions directly impacting energy efficiency in both fields: data storage, network transmission, task scheduling, model training, inference, and data pipelines.

The classification framework draws on insights from current research, identifying the critical factors that influence energy consumption at each process stage. It starts with big data analytics, exploring energy consumption across distributed storage systems, network transmission, and task scheduling. We also highlight emerging strategies for optimizing data storage, minimizing unnecessary data transfers, and improving task scheduling to promote more efficient energy usage.

For machine learning, the framework examines the energy implications of model training, inference, and the data processing pipeline. It delves into key optimization techniques such as model miniaturization, efficient GPU/TPU utilization, and data pipeline optimizations that help minimize computational and memory overhead. This holistic view illustrates how these strategies interact within cloud infrastructure to lower energy consumption while maintaining system performance.

By categorizing the energy consumption characteristics and optimization techniques within these two domains, we provide a structured approach to understanding how energy efficiency can be enhanced throughout the entire lifecycle of big data analytics and machine learning. This framework sets the stage for the following chapters, where specific optimization strategies and their limitations will be discussed in greater detail.

3.1.1 Taxonomy of Energy Consumption Characteristics

Chapter 4 focuses on the energy consumption characteristics of data storage, network transmission, and task scheduling in big data analysis, as well as model training, inference, and data pipelines in machine learning.

For big data analysis, it is essential to understand how different layers of cloud computing architecture influence energy efficiency. Given the complexity and energy demands of big data analytics, the characteristics should encompass multiple aspects, from data storage to computational resource scheduling. Therefore, this study adopts a framework based on three key dimensions: data storage, network transmission, and task scheduling and computation (Figure 1[2]).

As a critical component of big data analytics, data storage influences data persistence and access speed. In distributed cloud storage architectures, data redundancy replication and distribution significantly impact energy consumption [23]. For example, while multi-replica storage improves fault tolerance, it increases the energy consumption of storage devices [20]. Therefore, understanding the energy consumption characteristics of storage architectures and exploring how to reduce energy usage by optimizing resource management and storage structures has become a key area of research. Additionally, the scalability and dynamic adjustment of resources in the cloud environment also affects energy consumption, necessitating further analysis at this level [20].

Next, network transmission is another important factor influencing energy efficiency in big data analytics. Data typically needs to be transferred from remote servers to computation nodes during big data analysis. In distributed environments, the transmission cost increases exponentially with the volume of data [55]. Optimizing data transmission in cloud environments, significantly reducing the need for remote data transfers, has become a research focus. In this chapter, we explore applying local caching techniques, which can effectively reduce cross-region data transmission and thus lower energy consumption. We conclude by comparing these techniques with remote transmission, providing theoretical support for subsequent research model selection.

Finally, task scheduling and computation, as core mechanisms affecting energy efficiency, directly determine resource utilization efficiency and load distribution. Efficient task scheduling can prevent resource idling and reduce energy consumption during computation. Furthermore, appropriate resource integration and load balancing strategies in task scheduling can achieve a more balanced distribution of energy efficiency, preventing local overload or resource waste [19]. Therefore, we treat this as the final research focus in big data analytics, investigating how intelligent scheduling, resource integration, and real-time energy monitoring can further optimize cloud computing energy efficiency.

For machine learning, we have chosen three key dimensions: model training, inference phase, and data pipeline. These three processes typically significantly impact energy efficiency throughout the machine learning lifecycle (Figure 2).

First, the model training process in machine learning generally involves a substantial amount of computation, especially for deep learning models, which require intensive matrix operations and data processing, placing high demands on energy consumption [58]. Therefore, optimizing the computational complexity, memory access, and data transmission during the training process is a primary focus of this study. By exploring optimization strategies for these aspects, energy consumption during training can be significantly reduced. Additionally, the efficient use of hardware and parallel processing configurations can accelerate the training process and further improve energy efficiency [28].

The inference phase is another critical step in machine

[2]Influencing factors: The factors that mainly affect energy consumption at the current stage (including algorithm factors, technical factors, environmental factors, etc.)

learning systems, especially large-scale deployments. Efficient hardware resource utilization and choosing an appropriate batch size directly impact energy efficiency during inference [22]. The efficiency of hardware usage, such as the adaptation of GPUs and TPUs, is also a key factor affecting energy efficiency [42]. Therefore, this phase discusses batch size, hardware usage efficiency, and architectural differences as key aspects.

The data pipeline, which serves as the pathway for data in machine learning, plays a central role throughout the process. Data collection, cleaning, and transformation are often the most time-consuming and energy-intensive parts of a machine learning project [26]. How efficiently data ingestion, preprocessing, and feature transformation are performed significantly influences the overall energy efficiency of the machine learning process. This is particularly critical when handling large-scale datasets, which introduces challenges such as optimizing the data pipeline's scalability and real-time processing capabilities. For real-time processing, the challenge becomes how to reduce unnecessary data transmission and processing overhead while maintaining model performance. These issues are identified as key problems [26], which are thoroughly examined in the relevant literature, with corresponding solutions provided in this chapter.

The selection of these three dimensions is not only based on the practical energy consumption needs of each stage of machine learning but also considers their interconnections within the cloud computing architecture. Optimizing computation and memory usage during model training, optimizing hardware resource usage during the inference phase, and optimizing data processing efficiency in the data pipeline—these processes work together to influence the overall energy efficiency of the machine learning system. By thoroughly investigating these three areas, this study provides a comprehensive understanding of energy consumption characteristics in machine learning within cloud environments. It offers specific theoretical support and practical strategies for optimization.

3.1.2 Taxonomy of Optimization Strategies and Limitations

Chapters 5 and 6 focus on the optimisation strategies and limitations related to energy consumption in big data analytics and machine learning.

When investigating optimisation strategies and their limitations in big data analytics and machine learning, the primary consideration is minimising energy consumption while ensuring system performance. Based on the understanding of energy consumption characteristics in each domain presented in Chapter 4, we have designed an optimisation plan (Figure 4) grounded in an in-depth analysis of energy consumption features and potential bottlenecks in the optimisation process.

For big data analytics, the primary focus is on the energy consumption issues related to distributed storage systems. In big data applications, data storage is the most fundamental and core component, often requiring processing large volumes of data while ensuring data availability and reliability. Through extensive literature research, we identified the key optimisation strategies in distributed storage systems, which include improving fault tolerance and system reliability. Among these are data replication and erasure coding, which ensure data security and increase storage redundancy and computational overhead [59]. Thus, it is essential to evaluate their impact on energy consumption carefully [59]. Meanwhile, emerging green storage solutions such as GreenHDFS, which reduce unnecessary energy consumption through intelligent resource scheduling and dynamic power management, are incorporated as a key research area [21]. Additionally, we found that in distributed storage data consistency management and energy balancing, reducing unnecessary energy consumption—especially during data consistency and synchronisation processes—remains a technical challenge [43]. However, it is a necessary step toward optimising energy efficiency in big data systems. Therefore, relevant optimisation strategies are discussed in the corresponding section.

Furthermore, reasonable data partitioning and replication strategies are often required to enhance data processing efficiency and the system's fault tolerance. However, excessive partitioning and replication can lead to significant storage and network overhead, a trade-off widely discussed by researchers [2, 10]. Therefore, using this strategy to improve efficiency while reducing energy consumption is another key focus of this paper. The two strategies mentioned above form the foundation of optimising big data storage systems to ensure an efficient and energy-controllable data management approach.

In machine learning, deep learning models require substantial computational resources during training, and the inference process often faces challenges in optimizing hardware resources [56, 60]. To address these challenges, model miniaturization (knowledge distillation and pruning) has become one of the most mainstream optimization strategies [2]. Model miniaturization reduces models' scale and computational complexity, thereby lowering energy consumption [56]. This strategy reduces the demand for computational resources and significantly improves energy efficiency while maintaining performance. Pruning, as a method for reducing the complexity of neural networks, removes redundant neurons, further optimizing the model structure and improving computational efficiency [35].

Next, through extensive literature review, we observed that during deep learning training and inference phases, hardware resources such as GPUs and TPUs are often not fully utilized [60]. This observation led us to research the hardware utilization efficiency in the machine learning domain. As a result, GPU resource optimization was chosen as the second key optimization strategy. Many studies have found that adequately scheduling hardware resources can accelerate com-

putation and improve overall performance without increasing additional power consumption. This is because optimizing hardware utilization can reduce idle times and lower energy consumption, particularly in cloud environments, where resource sharing and scheduling are crucial for energy efficiency [50].

Lastly, as mentioned in Chapter 4, the data pipeline in machine learning, with its rich energy consumption characteristics, becomes a central area of our research on energy optimization strategies. We found that during large-scale data processing and feature transformation, energy consumption increases exponentially [24]. However, fine-tuned batch processing, data loading strategies, and data augmentation techniques can significantly reduce processing time and resource consumption [25, 32]. These strategies aim to optimize data flow and processing efficiency, thereby minimizing unnecessary computational and storage overhead.

In summary, the storage optimization, data partitioning, and replication strategies in big data analytics, as well as the model miniaturization, hardware optimization, and data pipeline management in machine learning, are all designed to reduce energy consumption and improve performance. Furthermore, these strategies complement each other to collectively enhance overall energy efficiency. We can provide more comprehensive and detailed theoretical support for future energy efficiency optimization by thoroughly examining these strategies and their limitations.

3.1.3 Future Directions of BDA and ML

Chapters 7 and 8 present a hierarchical framework (see Table 3), beginning with physical infrastructure, progressing through green energy adoption, and ultimately exploring intelligent resource management. This approach discusses the future directions for optimizing energy consumption in cloud computing. First, advancements in physical infrastructure, including technologies like liquid cooling, waste heat recovery, and specialized accelerators (e.g., TPUs and FPGAs), focus on reducing power consumption at the hardware level.

Next, integrating renewable energy is possible through AI-driven demand forecasting and multi-region load balancing, which help align computational workloads with sustainable power sources. The third area of focus involves AI-powered dynamic scheduling and cloud-native architectures, which optimize resource allocation using techniques like predictive analytics, Dynamic Voltage and Frequency Scaling (DVFS), and adaptive workload splitting.

Finally, model-centric strategies—such as pruning, quantization, and split computing—help to reduce computational overhead, akin to the data-centric optimizations found in big data analytics (e.g., distributed storage and in-memory processing). Together, these strategies highlight the importance of cross-layer collaboration, where innovations in hardware, energy systems, and algorithmic efficiency converge to address scalability and sustainability challenges. The overlap between BDA and ML emphasizes shared principles like workload distribution, adaptive resource management, and green-aware infrastructure, forming a roadmap for energy-conscious cloud computing.

The chapters provide full answers to the questions raised in RQ3 by conducting a comprehensive review of the literature and analyzing current trends.

4 Energy Consumption Characteristics

The rapid growth of cloud computing has significantly influenced the landscape of big data and machine learning, positioning them as critical components in contemporary data analysis and decision-making processes. However, as these systems scale, their energy demands have increasingly become a focus of discussion — especially in cloud environments where resource utilization is often associated with operational costs and environmental impact. This chapter examines the energy consumption characteristics of big data and machine learning applications in the cloud through key operational dimensions, including data storage, query optimization, transmission, task scheduling, and computational frameworks. Existing studies suggest these domains may present distinct challenges and opportunities for improving energy efficiency while requiring careful consideration of system performance and reliability constraints.

4.1 Data Storage in Big Data Analytics

The rapid growth of big data analytics has highlighted challenges in traditional centralized storage systems, including performance bottlenecks and single points of failure [23]. Distributed storage systems (DSS) have emerged as a widely adopted alternative, designed to provide scalability, fault tolerance, and high availability. By distributing data across multiple nodes, DSS can reduce access latencies and improve resource utilization, addressing the challenges of modern data workloads [23].

However, these advantages are often accompanied by increased energy consumption, particularly in large-scale cloud environments [23]. As a result, improving the energy efficiency of DSS has become a key priority for researchers aiming to balance performance with sustainability.

Observation (O-2): Energy consumption in big data analytics is primarily influenced by factors such as distributed storage systems, data transmission, and task scheduling. Specifically, the energy costs associated with data redundancy in distributed storage and data transmission across long distances contribute significantly to the overall energy usage. Optimizing storage architectures, minimizing data movement, and improving task scheduling strategies can substantially reduce energy consumption within big data systems.

4.1.1 Energy Consumption Trade-Offs in Distributed Storage Architectures

Distributed storage systems (DSS) support big data analytics in cloud environments. GlusterFS and Compuverde are often recognized for their scalability and fault tolerance, typically achieved through data replication. However, these benefits are generally accompanied by increased energy consumption [23]. For instance, Amazon Web Services (AWS) manages petabytes of data for many customers. Some of their large-scale data storage setups use GlusterFS for specific data types, such as log files and static content. According to data from 2023, in a particular AWS region where over 10 petabytes of data are stored using GlusterFS, the energy consumption due to data replication for redundancy purposes accounts for about 25% of the total storage-related energy usage [1].

When compared to Compuverde, in a similar data-intensive environment, a financial institution that tested both systems found that while Compuverde offers enhanced data redundancy through advanced erasure coding techniques, it tends to consume about 50% more energy than GlusterFS [23]. This is because the erasure coding and decoding processes in Compuverde are computationally intensive, requiring more CPU cycles and, therefore, more power. For example, when storing a dataset of 1 petabyte with a replication factor of 3 in GlusterFS, the annual energy consumption was measured at around 100,000 kWh. In contrast, using Compuverde with the same dataset and a comparable level of data protection through erasure coding, the annual energy consumption rose to approximately 150,000 kWh [23]. This highlights the potential trade-offs between advanced functionality and energy efficiency, emphasizing the need for architectures that balance these competing demands.

Replication strategies in distributed frameworks like Hadoop add another layer of complexity. While replication ensures data availability, it also increases energy consumption, particularly when redundant copies of infrequently accessed data remain idle. Optimizing replication frequency and data placement is crucial to minimizing energy overhead while maintaining system reliability [7].

Observation (O-3): Distributed storage systems like GlusterFS and Compuverde exhibit significant trade-offs between fault tolerance and energy efficiency. While erasure coding in Compuverde improves redundancy, its computational complexity increases energy consumption by 50% compared to replication-based systems like GlusterFS.

4.1.2 Replication and Resource Scaling in Data Centres

Data replication is an important strategy for improving data access efficiency and reducing network latency in data centers. In the context of cloud computing data centers, communication resources can often become a bottleneck for service provisioning in cloud applications, and data replication can help bring data closer to consumers, potentially minimizing network delays and bandwidth usage [7]. Dejene Boru et al. proposed a data replication technique that aims to jointly optimize energy consumption and bandwidth capacity [7]; their approach considered both energy efficiency and bandwidth consumption, along with improving Quality of Service (QoS) through reduced communication delays. They developed a replica manager at the central database, which periodically analyzes data access statistics to identify suitable data items for replication and their optimal sites. This approach sought to minimize data center energy consumption while attempting to maximize the available bandwidth in uplink and downlink transmissions [7].

Similarly, in distributed big data systems, data replication can help reduce long-distance data transfers by making frequently accessed data available locally [2]. Strategic replication across nodes can assist in avoiding excessive energy consumption related to data movement across the network [2]. For instance, in systems like Apache Hadoop and Spark, data replication techniques are often used to enhance data access speed and reduce energy consumption during query processing [2].

Resource scaling is another crucial aspect of data center management. Akhtar highlighted the concept of resource elasticity, where resources are allocated based on workload demand. Cloud platforms such as AWS, Google Cloud, and Azure typically support resource elasticity through autoscaling features, which allow for the deallocation of idle resources during low-demand periods to help save energy. This approach ensures that only the necessary resources are utilized, potentially optimizing energy consumption [2].

However, implementing replication and resource scaling strategies in data centers also presents challenges, including the complexity of implementation, which may require significant changes to existing systems and processes, resulting in high resource investment and initial costs [2]. Data center infrastructure variability can also influence the implementation of these strategies, and scalability issues might arise due to limitations in network bandwidth, storage capacity, and computational power [2].

4.2 Network Transmission Energy Characteristics in Big Data Analytics

Efficient network transmission is a critical aspect of energy consumption in big data analytics within cloud environments. In this section, we explore the trade-offs between remote data transmission and local caching strategies and analyze their impacts on energy efficiency in big data applications.

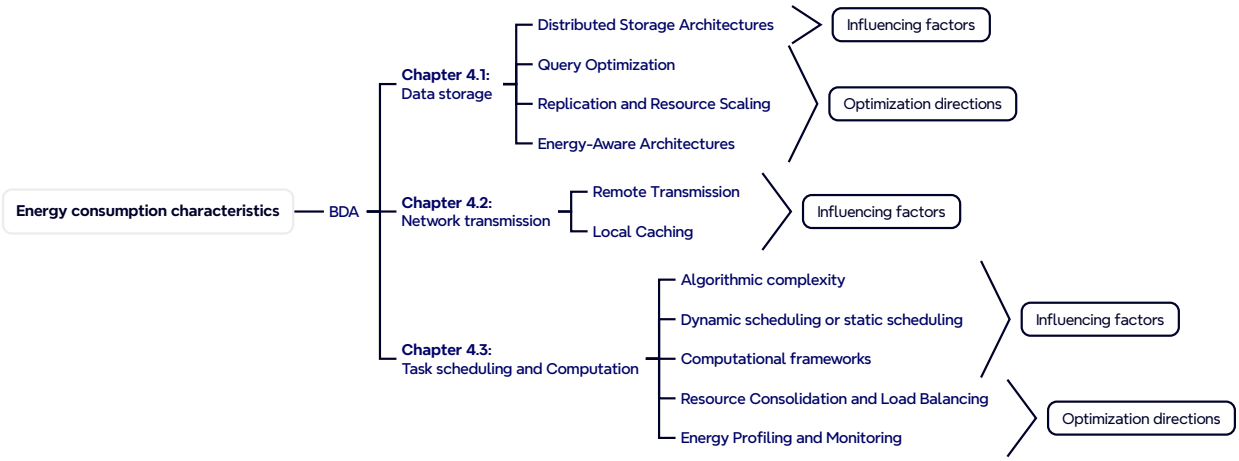


Figure 1: Energy consumption characteristics of BDA

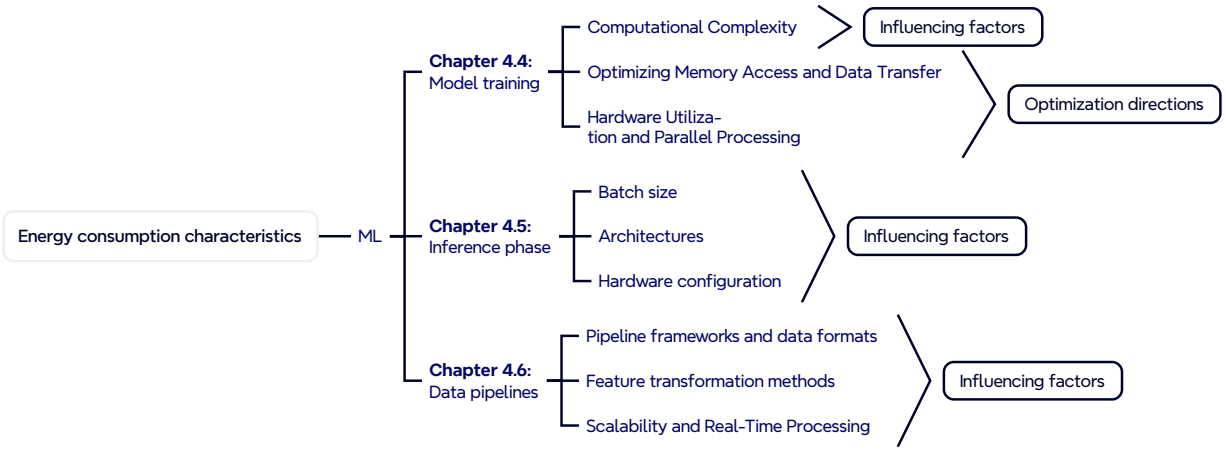


Figure 2: Energy consumption characteristics of ML

4.2.1 Energy Costs of Remote Data Transmission

Remote data transmission incurs substantial energy costs due to the involvement of multiple network components. Data transferred between users and cloud servers traverses access networks, core routers, and data centre switches, each contributing to the overall energy cost [55]. The energy consumption per bit for these components depends on their utilization rates. Underutilized components tend to exhibit higher energy usage per bit due to idle power requirements [55].

Packet latency and data volume are key factors influencing the energy efficiency of remote transmission. High-latency connections, such as those spanning wide-area networks (WANs), substantially increase energy costs compared to local processing. For instance, offloading database workloads to remote high-performance servers over high-latency networks led to a tenfold increase in energy consumption compared to local execution [37].

4.2.2 Local Caching

Local caching mitigates the need for frequent remote data transmission by storing frequently accessed data near computation nodes, this reduces network energy overhead and improves data retrieval times [37]. Experiments with database systems like MongoDB and Redis demonstrate that local caching significantly enhances energy efficiency for read-intensive workloads [37]. For example, repeated access to the same data blocks under low-latency conditions improved energy efficiency by up to 1.9 times in MongoDB scenarios [37].

Observation (O-4): Local caching reduces energy con-

sumption by up to 1.9× for read-intensive workloads in database systems, demonstrating its critical role in minimizing network transmission overhead in cloud-based big data analytics.

4.2.3 Comparative Analysis of Transmission Modes

The energy efficiency of local versus remote data access is influenced by workload characteristics and network configurations. Data-intensive tasks with minimal updates may be more suited to local caching, as the energy cost of transferring large datasets could outweigh the benefits of centralized processing. On the other hand, compute-intensive tasks, such as real-time data processing or large-scale analytics, may benefit from remote execution on high-performance servers, particularly when network latency is low [37, 55].

A key consideration in remote data transmission is encryption. Secure protocols like HTTPS can introduce additional energy costs due to encryption overhead, especially in applications that require frequent synchronization, such as collaborative cloud-based tools [55].

The trade-offs between remote transmission and local caching underscore the importance of aligning data access strategies with workload demands and network conditions. Striking a balance between these approaches could help optimize energy efficiency in cloud-based big data analytics [55].

4.3 Task Scheduling and Computation Energy Characteristics in Big Data Analytics

Efficient task scheduling and computational strategies aim to reduce the energy footprint while maintaining computational performance. This section explores how various task scheduling methodologies, computational frameworks, and resource utilization techniques contribute to energy efficiency.

4.3.1 Energy Implications of Task Scheduling

Task scheduling can have a significant impact on energy consumption by influencing how computational resources are allocated [36]. Heuristic and meta-heuristic algorithms, such as genetic algorithms and particle swarm optimization, have been identified as useful tools for balancing energy efficiency with performance metrics like makespan and resource utilization [36]. These algorithms adjust resource allocation dynamically based on real-time workload demands, potentially reducing energy waste while maintaining computational performance. By optimizing task distribution across resources, they may help minimize idle resource overhead and adapt to fluctuating system conditions [53].

In cloud computing environments, where workloads and resource availability are inherently dynamic, static scheduling strategies may not be sufficient [53]. Dynamic scheduling techniques are designed to address this challenge by continuously monitoring the status of virtual machines (VMs). When workloads become imbalanced, these systems can redistribute tasks from overloaded VMs to underutilized ones, helping to prevent energy waste caused by resource idling [2]. Additionally, preemptive scheduling could enhance energy efficiency during peak demand periods by allowing high-priority tasks to interrupt and reallocate lower-priority processes. This flexibility may enable systems to respond to sudden workload spikes while maintaining energy-efficient resource utilization [53].

Studies suggest that adaptive scheduling frameworks, which combine heuristic approaches with real-time monitoring, might reduce energy consumption by up to 35% compared to static methods in large-scale cloud environments [53]. Such advancements point to the potential benefits of intelligent, context-aware scheduling in promoting sustainable computing practices.

Observation (O-5): Adaptive scheduling frameworks combining heuristic algorithms and real-time monitoring reduce energy consumption by up to 35% compared to static methods, highlighting the importance of dynamic resource allocation in cloud environments.

4.3.2 Computational Energy Costs in Big Data Frameworks

The choice of computational frameworks can influence energy efficiency [36]. For example, Apache Hadoop, which is based on disk-based processing, is generally considered to consume more energy compared to Apache Spark, which uses an in-memory processing approach. Apache Spark has been reported to achieve energy savings of up to 40% for iterative workloads [36]. This improvement is often attributed to its reduced I/O operations and faster data processing capabilities, which may contribute to a more efficient system overall [36].

Additionally, task granularity can impact energy consumption. Fine-grained tasks, which require frequent communication between tasks, are typically associated with higher network energy usage compared to coarse-grained tasks [53]. This suggests that selecting an appropriate level of task granularity could play a role in optimizing energy efficiency. Specifically, aligning workload characteristics with the most suitable computational framework may help minimize energy expenditure [53].

Beyond choosing the right computational framework, other strategies, such as resource consolidation and load balancing, may also play a significant role in enhancing energy efficiency within cloud environments. Ensuring that computational resources are allocated effectively and balanced across the system can help reduce overall energy consumption while maintaining optimal performance.

4.3.3 Resource Consolidation and Load Balancing

Resource consolidation techniques, such as virtual machine (VM) migration, can be effective in reducing energy consumption within big data analytics environments. Organizations may potentially shrink their energy footprint by concentrating workloads on fewer active VMs and deactivating idle resources. Research suggests that well-executed workload consolidation in distributed systems could lead to energy savings of up to 25% through more efficient resource allocation [2]. While VM migration can incur an initial energy overhead, the long-term reductions in redundant resource usage may offset this cost.

Load balancing works in conjunction with resource consolidation by distributing tasks more evenly across available resources, helping to avoid situations where a VM becomes underutilized or overloaded. Modern load balancing algorithms have shown the potential to improve energy efficiency by 15-20% through real-time task redistribution [53]. For instance, comparisons between Apache Spark and Hadoop clusters suggest that effective load balancing can enhance system performance and potentially reduce energy use, particularly under high computational demand [36].

Integrating resource consolidation with energy profiling tools and adaptive algorithms may further improve energy efficiency. These tools enable dynamic workload adjustments, which help ensure that energy-saving measures evolve in response to changing conditions [2].

4.3.4 Energy Profiling and Monitoring

Energy-aware scheduling frameworks enable real-time tracking of energy metrics, which can facilitate timely adjustments to minimize energy wastage. These frameworks often leverage tools such as Intel's Power Gadget and customized monitoring solutions to analyze power consumption at the query, node, and cluster levels [2].

Potential energy savings can be realized by integrating energy profiling with advanced scheduling strategies; dynamic resource scheduling may allow for the deallocation of idle resources during low-demand periods, potentially leading to overall reductions in energy consumption [53]. In cloud environments, adopting energy-aware load balancing could result in energy savings of up to 15% by prioritizing efficient nodes and reducing cross-node data transfers [53]. Similarly, feedback loops informed by real-time monitoring may help systems refine energy-saving strategies, contributing to continuous optimization.

4.4 Model Training in Machine Learning

Model training is one of the most energy-intensive stages in machine learning, requiring substantial computational resources in cloud environments. This section explores the energy implications of training processes, focusing on computational complexity, memory dynamics, and hardware utilization.

4.4.1 Computational Complexity and Its Impact on Energy Consumption

Training deep learning models involves iterative processes such as forward passes, backwards passes, and parameter updates. These operations require many floating-point operations (FLOPs), contributing to the high energy demands [67]. Convolutional neural networks (CNNs) rely heavily on matrix multiplications during backpropagation, with Multiply-Accumulate (MAC) operations accounting for nearly 30% of their total energy consumption [67].

The energy cost per FLOP can vary depending on hardware specifications and the arithmetic precision used. Training frameworks optimized for half-precision (FP16) arithmetic can achieve up to a 50% reduction in energy consumption compared to single-precision (FP32) operations while maintaining comparable model accuracy [4]. This reduction highlights the importance of optimizing computational resources to reduce energy consumption without sacrificing performance.

Observation (O-6): Training frameworks using half-precision (FP16) arithmetic achieve up to 50% energy savings compared to single-precision (FP32) operations, underscoring the need for hardware-aware algorithmic optimization in ML training.

4.4.2 Optimizing Memory Access and Data Transfer to Reduce Energy Costs

Memory access and data transfer are significant contributors to training energy costs. In CNN training, data movement between DRAM and processing units can account for as much as 70% of total energy consumption [67]. To address these costs, researchers have developed various techniques.

One promising strategy involves caching frequently accessed parameters in high-speed buffers, which reduces the need for repeated data transfers between memory and processors [4]. Another notable approach is gradient checkpointing, where intermediate states are recomputed during backpropagation to minimize memory overhead, thereby reducing memory and energy consumption during large-scale training [4].

4.4.3 Enhancing Energy Efficiency Through Hardware Utilization and Parallel Processing

The energy efficiency of training largely depends on the hardware used and its capacity for parallel computations. GPUs and TPUs, designed for high-throughput operations, are reported to achieve 10–15 TFLOPs per watt, compared to the

1–2 TFLOPs per watt typically delivered by CPUs [4]. This suggests that substantial energy savings can be achieved by using specialized hardware for training.

Effective workload distribution across multiple GPUs is also important for maximizing resource utilization and minimizing idle power consumption. Two common strategies for parallelism are data parallelism, where the training data is distributed across devices, and model parallelism, where the model parameters are split across devices. These strategies can significantly improve energy efficiency in large-scale training [44]. Local caching may also play a crucial role in enhancing energy efficiency during data access. Netflix, for instance, is a leading streaming service that utilizes edge caches deployed across regions like North America to reduce energy-intensive long-distance data transfers. When a user watches a 500MB episode of a popular TV series like "Stranger Things," tests suggest that energy consumption for data transmission from the primary data centre is around 100 joules [17]. With local caching, the same episode retrieved from the nearest edge server can reduce this consumption to just 20 joules [17]. Beyond the energy savings, local caching can also enhance streaming speed, potentially improving the overall user experience.

The above approaches, from optimized hardware utilization to efficient data access mechanisms, highlight the potential benefits of integrating specialized strategies to promote sustainable energy usage in computational systems.

4.5 Inference Phase in Machine Learning

Unlike training, which is highly computationally demanding, inference prioritizes efficient response times and low energy consumption. This section delves into the energy dynamics during inference, focusing on factors such as batch size, model architecture, and hardware utilization.

4.5.1 Energy Impact of Batch Size

Batch size influences the balance between energy efficiency and response latency during inference. Larger batch sizes tend to enhance hardware utilization, potentially reducing the energy cost per instance processed [39]. For example, when the batch size of models like AlexNet and ConvNext increases from 1–2 to 128, power consumption rises significantly, from 65 W to 87 W for AlexNet and from 93 W to 166 W for ConvNext (Table 1) [62]. While this scaling can improve hardware efficiency, it may introduce latency, particularly for workloads with low-frequency requests, suggesting a trade-off between energy efficiency and response time [39].

However, not all models show the same energy consumption trends as batch sizes change. ShuffleNetV2, for instance, appears to maintain consistent energy consumption across a wide range of batch sizes, whereas DenseNet seems to achieve optimal efficiency at smaller batch sizes, typically around 16 or fewer [62]. This indicates that model-specific characteristics can play a key role in determining the optimal batch size for energy efficiency during inference.

Model	Batch size 1–2 (W)	Batch size 128 (W)	Difference (%)
AlexNet	±65.0	87.3	±34.3
DenseNet	72.5	163.1	124.9
ShuffleNetV2	±65.0	87.1	±33.9
VisionTransformer	76.2	185.8	144.0
ConvNext	93.1	166.4	78.6

Table 1: GPU peak power in Watts (W) differences for small and large batch sizes [62].

Observation (O-7): Batch size significantly impacts inference energy efficiency, with larger batches improving hardware utilization but introducing latency trade-offs. Model-specific characteristics (e.g., ShuffleNetV2 vs. DenseNet) further dictate optimal batch configurations.

4.5.2 Hardware Utilization During Inference

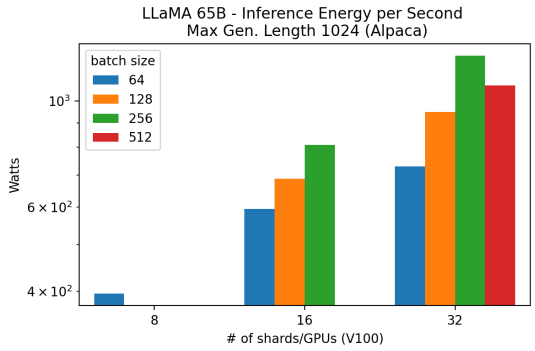
The hardware configuration used during inference can significantly influence the overall energy consumption of a model. Different GPUs tend to exhibit varying levels of energy efficiency depending on workload distribution. For example, inference of the LLaMA 65B model across 8 V100 GPUs showed a noticeable range in power consumption, from 300 W to 1000 W, depending on both the number of shards and the batch size (Figure 3) [39]. While increasing the number of shards can improve throughput, it may also result in higher energy consumption per decoded token and response, particularly for larger batch sizes [39].

More granular adjustments in hardware configuration may lead to improvements in energy efficiency [39], suggesting the potential of customizing hardware resources to suit specific inference tasks.

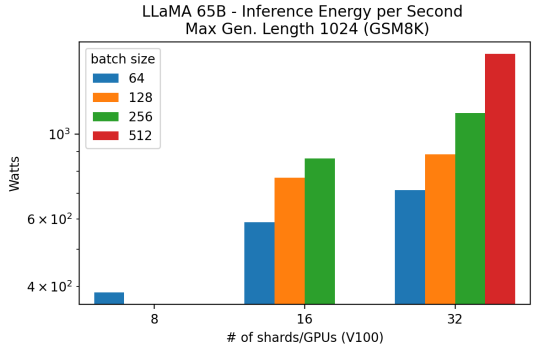
Observation (O-8): Optimizing hardware utilization, particularly with GPUs and TPUs, is crucial for improving energy efficiency in machine learning applications. Techniques such as kernel-level scheduling, pipeline parallelism, and dynamic container orchestration help ensure that resources are used effectively, reducing idle power consumption. These methods are key to making machine learning processes more energy-efficient, particularly in cloud-based environments where resource allocation is often dynamic.

4.5.3 Architectural Differences and Their Impact on Energy Efficiency

Neural network architectures can present diverse energy profiles during inference. Transformer-based models like VisionTransformer may tend to show more energy-efficient behaviors as batch size increases. This is likely due to the parallelizable nature of transformers, which allows for better scalability,



(a)



(b)

Figure 3: Inference Energy Estimates of LLaMA 65B for Varying Batch Sizes and Shard Numbers [39]

potentially reducing the energy cost per instance while maintaining faster inference speeds [62].

In contrast, convolutional neural networks (CNNs) like AlexNet and DenseNet often show a different energy consumption profile. These models might be more energy-efficient when smaller batch sizes are used, likely due to the less parallelizable nature of convolution operations and the high memory access costs involved [62]. Architectural design therefore needs to be considered when selecting batch sizes to optimize energy efficiency.

Moreover, the complexity of the model can further influence these architectural effects. Larger models, such as the LLaMA variants (7B, 13B, and 65B), appear to exhibit an exponential increase in energy consumption per second as both the GPU shard count and batch sizes scale up [39]. As models become more complex, the demand for optimizing energy-efficient architectures may continue to increase.

4.6 Data Pipelines in Machine Learning

Data pipelines facilitating the movement and transformation of data from raw inputs to model-ready formats. This section explores the energy dynamics involved in the key stages of data pipelines.

Observation (O-9): Optimizing the data pipeline is essential for reducing energy consumption during machine learning tasks. Efficient data ingestion, preprocessing, and feature transformation can lower computational and storage overhead significantly. These improvements are particularly critical when dealing with large-scale datasets, as they help minimize energy costs while maintaining model performance, especially for real-time processing tasks.

4.6.1 Energy Costs of Data Ingestion and Transformation

The energy consumption associated with data ingestion and transformation processes highly depends on the choice of pipeline frameworks and data formats [18]. Tools like Apache Kafka and Telegraf, commonly used for real-time data processing, exhibit notable energy efficiency variations based on how they are configured. When used for time-series data, they process 10,000 records in an average of 0.39 seconds, increasing to 1.26 seconds for 100,000 records. This performance boost translates into a substantial reduction in energy consumption, outperforming traditional Python-based implementations by up to 90 times in speed and corresponding energy savings [18].

Another key factor influencing energy usage is the data format itself [40]. While JSON and CSV formats are widely used for their human readability, formats such as HDF5, designed for hierarchical data storage, offer superior energy efficiency when handling large-scale datasets. This is mainly due to their reduced input/output (I/O) overhead, making them more suitable for big data applications. As shown in Table 2, HDF5 format significantly reduces energy consumption compared to JSON, with Pandas (PKG) consuming 82.2% less (565 vs. 3178.7), Dask (PKG) consuming 85.5% less (502.7 vs. 3471.9), and Vaex (PKG) consuming 8.6% less (3379.2 vs. 3697.8), making it a more efficient choice for dataframe input [40].

4.6.2 Preprocessing and Feature Transformation

Energy consumption during preprocessing operations, such as data cleaning, handling missing values, and performing feature transformations, can vary across different frameworks. Libraries like Vaex, known for their memory-mapped processing and lazy evaluation approach, are reported to consume up to 202 times less energy than Pandas when processing large datasets [40]. This significant difference is often attributed to Vaex's ability to process data without loading everything into memory simultaneously, which may help reduce energy demand.

On the other hand, though popular for their ease of use, Pandas tend to exhibit higher energy costs, particularly for

File format	Adult Dataset			Drugs Review Dataset		
	Pandas (PKG)	Vaex (PKG)	Dask (PKG)	Pandas (PKG)	Vaex (PKG)	Dask (PKG)
CSV	551.2	1030.4	62.6	9696.6	13042.2	65.1
JSON	3178.7	3697.8	3471.9	156434.5	160102.2	154388.9
HDF5	565	3379.2	502.7	5568.2	4272.1	NA

Table 2: Mean values for energy consumption for dataframe input in different formats [40].

tasks that require iterative data manipulation [40]. In these cases, energy consumption may increase as the dataset grows, suggesting the importance of selecting appropriate tools for preprocessing tasks [40].

In addition to these general optimizations, feature transformation methods like normalization and one-hot encoding can also influence energy efficiency [36]. Apache Spark, which utilizes in-memory computation and parallel processing, has been found to significantly reduce processing time and energy costs compared to traditional, single-threaded approaches. This efficiency gain is often more pronounced when processing large, distributed datasets [36].

4.6.3 Scalability and Real-Time Processing

Well-designed distributed processing systems allow pipelines to scale with minimal additional energy cost [18]. Doubling the input data size in a highly optimized pipeline typically results in an energy increase of less than 10%, whereas less optimized systems can experience energy consumption rises exceeding 30% [18].

Real-time data processing introduces another layer of complexity. Systems incorporating tools like Kafka and InfluxDB can balance low latency and high throughput. However, these systems face significant challenges when dealing with fluctuating traffic patterns [66]. Under steady workloads, they operate efficiently with minimal energy consumption. In contrast, spikes in data flow or unpredictable workloads can lead to disproportionately higher energy usage, as the system must adapt to varying demand levels [18, 66].

5 Optimization Strategies in Big Data Analytics

Big data analytics demands effective strategies to manage large-scale data processing, storage, and management complexities. As data volume, variety, and velocity continue to grow, optimization strategies become essential to ensure that big data systems can scale efficiently, all while maintaining performance, reliability, and energy efficiency.

This chapter covers strategies for enhancing different aspects of distributed big data systems, such as fault tolerance, energy efficiency, data partitioning, replication, and consistency management. Energy-efficient techniques, like zoning architectures and dynamic data placement, refine energy consumption by adapting to access patterns and minimizing resource usage.

Overall, these optimization strategies balance big data systems' performance, energy efficiency, and fault tolerance. However, they also come with trade-offs and limitations, including increased complexity, potential performance degradation under heavy workloads, and the need for dynamic adjustments to accommodate evolving data access patterns. Therefore, carefully considering these factors is necessary when designing and implementing optimization strategies for big data analytics.

5.1 Reliable Distributed Storage Systems

Distributed storage systems are fundamental to the scalability and reliability of big data analytics, enabling efficient storage and processing of large-scale datasets. Distributed storage systems, such as HDFS, Cassandra, and GlusterFS, employ various optimization strategies to address fault tolerance, scalability, and energy efficiency.

5.1.1 Fault Tolerance and Reliability

Distributed storage systems employ replication and erasure coding to ensure reliability and fault tolerance. HDFS and Cassandra use replication, storing multiple copies of data across nodes to minimize the risk of data loss caused by hardware failures [9]. In contrast, systems like Ceph and GlusterFS rely on erasure coding, which splits data into fragments and adds parity bits, offering fault tolerance with reduced storage overhead compared to replication [23].

5.1.2 Energy-Efficiency Strategies

GreenHDFS enhances energy efficiency by introducing a zoned architecture that categorizes data into "hot" and "cold" zones based on access frequency. Hot data, accessed frequently, is placed on active servers for faster retrieval, while cold data is stored on idle servers that operate in power-saving modes. A simulation of Yahoo's Hadoop clusters demonstrated a 26% reduction in energy consumption for the cold zones [21].

Another advanced strategy involves dynamic data placement. Instead of using fixed classifications, this approach con-

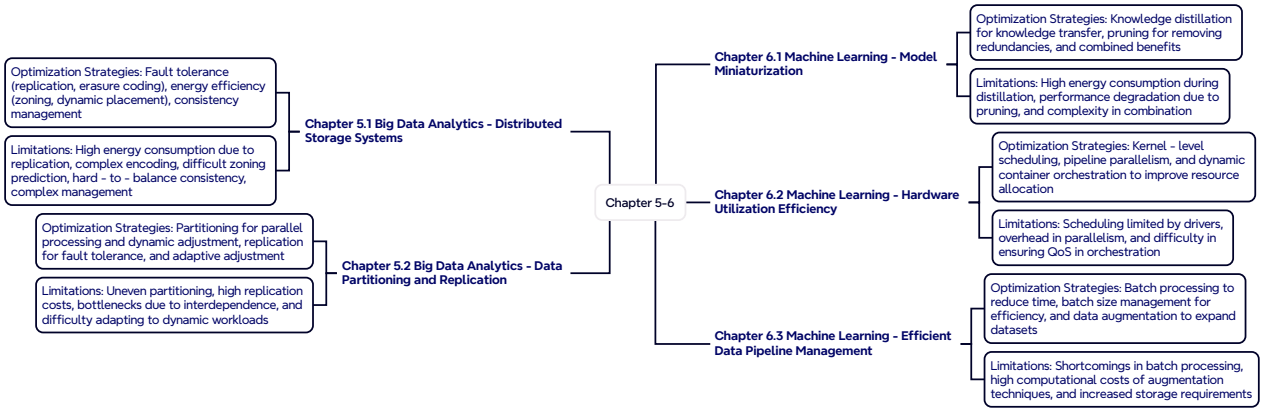


Figure 4: Optimization strategies and limitations in both fields

tinuously analyzes access patterns and adjusts real-time data placement. For instance, if a piece of data initially classified as cold starts to receive more requests, it can be dynamically moved to a more active and power-efficient node. This helps conserve energy and optimizes the response time by ensuring that frequently accessed data is always close to where it is needed most [43].

5.1.3 Consistency Management and Energy Balancing

Cassandra introduces tunable consistency levels, which allow for a trade-off between energy usage and data consistency. Eventual consistency reduces energy consumption by propagating updates asynchronously, although this can result in serving stale data. On the other hand, strong consistency ensures real-time synchronization but increases energy consumption [9].

Some systems use hybrid consistency models to address the limitations of eventual consistency [43]. These models allow for flexibility by adjusting the level of consistency based on specific use cases or data requirements. Critical data might require strong consistency to ensure all nodes are synchronized in real-time, while less critical data can remain inconsistently synchronized to save energy.

5.1.4 Limitations

Despite these advancements, distributed storage systems face several limitations.

Replication, while ensuring high fault tolerance, substantially increases storage overhead and energy consumption, particularly in systems dealing with large datasets. Although erasure coding offers a more storage-efficient alternative, it incurs computational complexity during the encoding and decoding processes, which leads to higher CPU usage and latency [23, 43].

Energy-efficient zoning, such as the one used in GreenHDFS, depends heavily on the accuracy of predictions regarding data access patterns. Inaccurate classification of frequently accessed data as cold can lead to significant performance degradation, as retrieving such data from cold zones results in higher latencies [21]. Moreover, systems need sophisticated mechanisms to predict and adapt to changing access patterns in real-time, which is not always feasible.

In systems like Cassandra, the trade-off between consistency and energy consumption is not always straightforward. Strong consistency guarantees up-to-date data but increases latency and energy use due to synchronous operations. Eventual consistency, on the other hand, reduces energy consumption but risks delivering outdated or inconsistent data to users [9].

Finally, the integration of multiple optimization techniques adds complexity to system management. Balancing energy efficiency, fault tolerance, and performance requires careful configuration, frequent monitoring, and constant adjustments, all of which increase operational costs and maintenance efforts.

Observation (O-10): Distributed storage systems face a trilemma between energy efficiency, fault tolerance, and performance. Techniques like erasure coding and dynamic zoning reduce energy costs but introduce computational overhead and management complexity, requiring context-aware configurations.

5.2 Data Partitioning and Replication

Data partitioning and replication are key techniques in distributed big data systems. They aim to optimize data locality, reduce network overhead, and improve energy efficiency, especially in cloud environments.

5.2.1 Partitioning

Partitioning divides large datasets into fixed-size blocks that are distributed across nodes in a cluster. This method enables parallel processing, reducing processing latency and network communication by localizing data access [2]. Improves system performance by redistributing data according to real-time access patterns. This helps address issues like data hotspots and load imbalances, ensuring resources are used more efficiently. In real-time stream processing environments, such as Storm, dynamic partitioning adjusts the distribution of workloads across nodes to prevent bottlenecks and optimize resource utilization [10].

5.2.2 Replication

Replication works alongside partitioning to ensure fault tolerance and quick data retrieval. In systems like Apache Cassandra, tunable consistency levels allow users to balance energy efficiency with the desired level of data accuracy. For applications with less stringent requirements, eventual consistency can reduce the need for redundant updates, thereby lowering energy costs during write operations [2]. Adaptive replication strategies take it a step further, adjusting the replication factor dynamically based on workload intensity and node availability. This flexibility improves energy efficiency and resource utilization, especially in systems with fluctuating demand [49].

5.2.3 Limitations

While enabling efficient data localization, partitioning can lead to data skew and uneven workload distribution across nodes. This imbalance results in specific nodes becoming overburdened while others remain underutilized, adversely affecting the overall system efficiency and energy consumption [47]. Moreover, achieving optimal partitioning requires detailed knowledge of the workload and access patterns, which are often dynamic and challenging to predict in real-world applications [10]. As systems scale, the complexity of partition management increases, further complicating the task of maintaining balance across the nodes.

Replication enhances fault tolerance and availability by storing multiple copies of data across nodes. However, excessive replication introduces high storage costs and overhead, particularly in energy-constrained environments like edge computing or IoT scenarios [2]. Maintaining consistency among replicas in highly dynamic systems can also lead to significant synchronization overhead, increasing latency and energy consumption [138]. For example, maintaining strong consistency in distributed systems often involves complex protocols such as Paxos or Raft, which can degrade performance under heavy workloads [46].

Another limitation arises from the interdependency of partitioning and replication. In systems using both strategies, cross-node communication can become a bottleneck. Replicating frequently accessed partitions on multiple nodes reduces communication latency but increases storage and synchronization costs. Conversely, limiting replication to reduce costs can lead to more extended data retrieval when non-local data access is required, undermining the advantages of partitioning [2].

Finally, both strategies often fail to account for the dynamic nature of modern workloads. As data volume and access patterns evolve, static partitioning and replication configurations become less effective. While addressing these issues, dynamic reconfiguration mechanisms introduce additional complexity and runtime overhead, which can negatively impact system performance and scalability [2, 47].

Observation (O-11): While data replication is an important strategy for ensuring data availability and fault tolerance, it also introduces significant energy and storage costs, especially in dynamic workloads. The key challenge lies in optimizing replication strategies to balance energy consumption with data reliability. Dynamic adjustment of replication factors based on workload fluctuations can help reduce unnecessary energy expenditure while maintaining system robustness.

6 Optimization Strategies in Machine Learning

Optimizing machine learning systems within cloud environments requires a comprehensive strategy that balances computational efficiency, resource use, and sustainability. This chapter delves into three core pillars of optimization: model compression, hardware utilization, and data pipeline management. These approaches respond to the increasing demand for scalable, energy-efficient machine learning in resource-limited cloud infrastructures. However, implementing these strategies involves inherent trade-offs, such as potential performance loss, synchronization costs, and heightened computational complexity. This chapter outlines ways to balance efficiency improvements with environmental and operational sustainability in modern cloud-based ML deployments by critically examining these methods.

6.1 Model Miniaturization

Miniaturizing machine learning models has become a key strategy for optimizing performance while managing resource constraints in cloud computing environments. Knowledge distillation (KD) and pruning are two primary techniques that effectively reduce model size, latency, and energy consumption [57, 63]. When combined, these methods offer even more significant resource optimization. However, a deeper

exploration of their impact on energy savings is required to understand their potential fully.

Observation (O-12): Techniques such as knowledge distillation and pruning effectively reduce the energy consumption of machine learning models, particularly during the inference phase. By decreasing model size and computational complexity, these strategies lower energy requirements while maintaining accuracy. However, aggressive pruning can lead to performance degradation, particularly in tasks requiring highly complex models, which underscores the need to balance energy efficiency with model effectiveness.

6.1.1 Knowledge Distillation

Knowledge distillation (KD) compresses complex models by transferring the knowledge of a high-capacity teacher model to a smaller student model. This process typically involves aligning key aspects such as logits, attention maps, and hidden states. In the EfficientVLM model, KD reduced energy consumption during inference by 37% while preserving 98.4% of the teacher model's performance, demonstrating its substantial efficiency gains [56]. Similarly, KDGAN utilized adversarial training to accelerate inference speeds, resulting in a 20% reduction in computation costs [57, 63].

6.1.2 Pruning

Pruning is another complementary technique that enhances energy efficiency by removing redundant model parameters. Structured pruning, which targets components like attention heads and feedforward layers in transformer architectures, has proven especially effective in reducing energy consumption. Adaptive pruning in BERT led to a 25% reduction in energy use per inference task without sacrificing accuracy [12]. Similarly, modal-adaptive pruning in EfficientVLM achieved even greater improvements, doubling the inference speed while cutting down power consumption [56].

When combined, KD and pruning produce compounded benefits. In the "distil-then-prune" approach, the teacher model is first distilled to retain essential information, after which pruning eliminates redundancies, resulting in energy savings of up to 30% across language and vision tasks [12]. Alternatively, the "prune-then-distill" strategy simplifies the teacher model first, yielding a student model that consumes 40% less energy per inference than those derived from unpruned teachers [35]. By combining these two techniques, models become smaller and more energy-efficient, ensuring that cloud resources are utilized optimally.

6.1.3 Limitations

There is still much room for optimization in the application of KD and pruning. Knowledge distillation, while reducing model size and improving inference efficiency, demands additional computational resources during the distillation phase. Some studies on large-scale transformer models like BERT show that distillation can require up to 20% more energy during the transfer process compared to training a smaller model from scratch, which partially offsets the energy savings gained during inference [12].

Pruning also has its drawbacks. Aggressive pruning may result in performance degradation, particularly in tasks requiring high model complexity. Unstructured pruning, which selectively removes individual parameters, can lead to irregular memory access patterns that are not well-suited for hardware optimized for uniform data flows, thus increasing power consumption [12, 35]. Structured pruning may alleviate some of these issues in some specific model components. However, it can still cause accuracy drops in tasks such as semantic understanding, even with energy savings of up to 25% [12].

Integrating KD and pruning adds further complexity. The "distil-then-prune" approach might be hindered by redundancies introduced during the distillation phase, which could limit the potential energy savings. On the other hand, the "prune-then-distil" approach risks suboptimal knowledge transfer if the pruned teacher model lacks sufficient representational power. Experiments with ResNet models highlight this trade-off, where a 35% reduction in energy consumption was achieved at the cost of a 5% loss in accuracy [35].

Observation (O-13): While model miniaturization techniques like knowledge distillation and pruning reduce inference energy by up to 40%, their training-phase overhead (e.g., 20% extra energy for distillation) and accuracy trade-offs necessitate careful cost-benefit analysis.

6.1.4 Energy Savings Through Miniaturization

Empirical research underscores the energy-saving potential of combining KD and pruning. In transformer-based models, the integration of these techniques has led to a reduction in inference energy consumption by up to 40%, while training energy costs decreased by 20% compared to baseline models. EfficientVLM applied both distillation and pruning sequentially, reduced model size by 44.3% and cut power usage during inference by 37% [12, 56]. These findings demonstrate that model miniaturization strategies reduce the computational burden and help minimize the environmental impact of machine learning operations, making them highly suitable for cloud-based environments with stringent resource limitations.

6.2 Hardware Utilization Efficiency

Optimizing hardware utilization in machine learning systems focuses on improving the allocation of GPU and TPU resources using advanced scheduling and orchestration techniques. Techniques like kernel-level scheduling, pipeline parallelism, and dynamic container orchestration have proven

highly effective in boosting computational efficiency while helping to reduce energy usage [48, 60].

6.2.1 GPU Resource Optimization

Kernel-level scheduling frameworks, like FIKIT, boost resource efficiency by exploiting inter-kernel idle times to run auxiliary tasks. This approach has led to a 30% improvement in energy efficiency in real-time GPU multitasking scenarios [60]. Similarly, pipeline parallelism on TPUs allows for the simultaneous processing of different model segments, achieving up to a 12x speedup for complex models like ResNet-152 while lowering energy consumption per inference frame [48].

Dynamic container orchestration, exemplified by platforms such as Kube-Knots, mitigates resource fragmentation by dynamically resizing containers based on workload demands. This technique effectively taps into idle GPU cycles, reducing total cluster energy consumption by 33% without sacrificing performance [50].

Empirical data supports the energy-saving potential of these strategies. For instance, pipeline parallelism on TPUs reduced energy usage per inference frame by as much as 40% compared to single-TPU setups [48]. Kernel-level scheduling resulted in a 25% decrease in GPU idle times, providing significant energy savings for GPU-intensive machine learning tasks [60]. Additionally, dynamic container orchestration has been shown to cut overall energy consumption by 33% in GPU-powered data centres [50].

6.2.2 Challenges and Limitations

Kernel-level scheduling, such as the one used by FIKIT, faces challenges in determining optimal execution patterns, mainly due to the proprietary nature of GPU drivers. This limitation reduces its flexibility when adapting to various hardware configurations [60].

While pipeline parallelism speeds up model training and inference, it introduces significant overhead from model segmentation and parameter synchronization. This overhead can diminish the energy savings in multi-TPU setups, where the cost of synchronization might surpass the advantages of parallel processing [48].

Dynamic container orchestration also presents hurdles in maintaining Quality of Service (QoS) for latency-sensitive applications. Resource fragmentation and potential interference between shared GPU environments can lead to inconsistent performance, particularly when containers are dynamically resized to meet workload demands [50].

6.3 Efficient Data Pipeline Management

Properly managing data pipelines in machine learning supports both performance and scalability [25]. Strategies such as batch processing, loading, and data augmentation, along with their limitations, are discussed below, as highlighted in recent literature.

6.3.1 Batch Processing and Loading

Batch processing helps address the computational demands of machine learning workflows, reducing training times and alleviating resource bottlenecks. The Plumber framework, for example, achieved up to 47× speedups in misconfigured pipelines, delivering end-to-end improvements exceeding 50% compared to traditional tuning methods [24]. Analyzing over two million ML jobs, Plumber identified software inefficiencies, rather than hardware constraints, as the leading cause of underperformance, highlighting the importance of well-tuned configurations to maximize resource efficiency [24].

Managing batch sizes is another effective strategy for improving energy efficiency. In edge computing scenarios, increasing batch sizes boosts GPU utilization and reduces idle time. Some recent video analysis experiments on edge devices demonstrated a 25% reduction in idle energy consumption with larger batch sizes compared to smaller ones [25].

However, when dealing with heterogeneous datasets of varying sizes, traditional batch processing can lead to underutilization of hardware accelerators like GPUs [25]. In distributed systems, while data partitioning improves throughput, it often incurs significant communication overheads, limiting scalability [5].

6.3.2 Data Augmentation Strategies

Data augmentation is a technique used to artificially enlarge training datasets, which is especially useful when dealing with limited data. The main aim of data augmentation is to enhance the robustness and generalization of machine learning models by introducing variations in the training data. This helps the model learn a broader range of features, ultimately reducing the risk of overfitting. Standard data augmentation methods include geometric transformations like rotation, scaling, cropping, and flipping, as well as neural rendering and synthetic data generation using approaches such as Generative Adversarial Networks (GANs). Techniques like geometric transformations, neural rendering, and GAN-based synthesis diversify datasets, helping to reduce overfitting [27, 32]. For example, 3D modelling-based augmentation boosts generalization in vision tasks by adding realistic variability to the training data [32].

There are inevitable trade-offs that come with data augmentation technology. While effective at generating realistic data samples, GAN-based augmentation is computationally heavy and requires precise tuning of the network's parameters to strike a balance between fidelity and diversity [32]. The added computational load from these advanced techniques can significantly increase energy consumption, which is particularly

important in resource-constrained deployment environments. Furthermore, since augmented data may not perfectly capture real-world variations, excessive use of augmentation could lead to datasets that stray too far from actual distributions, potentially degrading model performance on rare or outlier cases.

Additionally, over-aggressive transformations can distort the original data distribution, resulting in biased models that perform poorly on edge cases or unusual data points [27]. On top of that, advanced data augmentation methods can create higher storage demands, as larger datasets need to be stored. This can be problematic for systems with limited storage, such as edge devices or mobile platforms.

6.3.3 Reflection

Combining optimized batch processing with advanced augmentation techniques can effectively address many performance bottlenecks. For example, adaptive adjustments to batch sizes help maximize resource utilization while minimizing delays, as demonstrated in edge computing applications [25]. However, striking the right balance between these optimizations and the complexity of modern datasets and distributed environments remains a significant challenge.

While these strategies offer notable improvements, their implementation requires careful orchestration to overcome limitations and ensure scalability.

7 Advancements in Energy-Efficient Big Data Analytics

Future data centre designs and analytics frameworks should carefully navigate the growing demand for computational power alongside pressing environmental objectives [31, 38]. This chapter adopts a layered perspective to show emerging and anticipated advancements in energy-efficient big data analytics, encompassing innovations in physical infrastructure, green energy adoption, intelligent resource management, data-driven optimization, and cloud-native architecture. Each layer holds the potential for significant research and breakthroughs in the coming years.

7.1 Physical Infrastructure and Hardware Strategies

Significant energy savings can often be achieved at the physical layer, where advanced cooling technologies and optimized data centre layouts are critical in reducing operational expenses and carbon emissions [31]. Future data centres are expected to integrate breakthroughs in thermal management, waste heat recovery, and intelligent facility design to address the demands of escalating computational workloads [31].

7.1.1 Enhanced Cooling Mechanisms and Data Centre Design

Enhancing cooling systems is a critical aspect of improving energy efficiency, and as demands grow, future data centres are expected to refine existing methods while adopting innovative approaches. For instance, free cooling—harnessing natural climatic conditions—has already demonstrated its ability to reduce energy consumption by up to 30% in colder regions [31]. Emerging designs will likely feature intelligent systems that monitor external temperatures and airflow patterns, allowing data centres to alternate between mechanical and natural cooling modes in real-time dynamically.

In Sweden, Ericsson-operated data centres have successfully implemented waste heat recovery systems, achieving a 15% reduction in cooling energy requirements by repurposing server-generated heat as usable energy for facility heating [31, 38]. Future developments will likely focus on more advanced heat exchange technologies, pushing the boundaries of thermal energy reuse to improve energy efficiency further.

Meanwhile, liquid cooling systems are positioned to gain traction as hardware densities continue to rise. By circulating specialized coolants directly over high-heat components, these systems can lower server temperatures by approximately 30% [31]. This approach significantly reduces the need for air conditioning, making it a promising solution for next-generation, high-density data centres. When combined with optimal layout designs, such as hot/cold aisle containment, these advancements are expected to drive operational efficiency even further in the years ahead.

7.2 Green Energy Integration

With sustainability and corporate responsibility becoming top priorities, adopting renewable energy and intelligent resource scheduling will take on an even greater role in future data centres, which can seamlessly adapt to the variability of green power sources like wind and solar [8, 38].

7.2.1 Transition to Renewable Energy Sources

Leading cloud providers such as Microsoft and Google are increasingly shifting toward renewable energy, a trend poised to accelerate as solar and wind technologies become more affordable. According to research by Buyya and Gill [8], integrating renewable energy sources can cut carbon emissions by as much as 50%. In the future, dynamic energy management systems are expected to evolve, enabling seamless real-time coordination between renewable and traditional energy sources [8, 38].

These advancements will also drive the development of more intelligent energy storage solutions, including cutting-edge battery systems, supercapacitors, and hydrogen-based storage technologies [8]. Such innovations will mitigate the

intermittency of solar and wind power, ensuring steady and efficient operation even during periods of low energy generation. Additionally, geographical load balancing—redirecting workloads to regions with the most favourable renewable energy conditions—will empower data centres to scale their reliance on green energy with greater efficiency [8].

Observation (O-14): Integrating renewable energy with AI-driven demand forecasting reduces data center carbon emissions by up to 50%, but geographic load balancing and advanced storage systems are critical to mitigating intermittency challenges.

7.3 Intelligent Resource Management

As workloads become more dynamic and complex, AI-driven solutions are trending toward proactively monitoring, predicting, and optimizing resource allocation [31]. In the coming years, machine learning algorithms are expected to advance to process larger data streams and respond to unpredictable usage patterns with almost real-time accuracy [31].

7.3.1 Adoption of AI-Driven Energy Management

AI-powered frameworks hold tremendous potential for revolutionizing energy management in data centres. Milić demonstrated the use of the OODA (Observation, Orientation, Decision, and Action) loop in conjunction with K-means clustering to optimize cooling systems based on data from over 90,000 monitoring points, achieving a 20% improvement in cooling energy efficiency [31]. Looking ahead, emerging systems are expected to leverage deeper neural networks and advanced anomaly detection, further automating facility controls and driving efficiency gains.

Next-generation AI is also poised to transform load balancing, dynamically allocating workloads to underutilized servers and enabling Power Usage Effectiveness (PUE) ratings as low as 1.2 in experimental scenarios [31]. Beyond cooling, predictive analytics and machine learning will extend their influence to predictive maintenance, allowing data centre operators to schedule repairs proactively and minimize downtime [31]. Continuous learning capabilities will empower these algorithms to adapt quickly, aligning energy consumption with fluctuating computational demands and user behaviours [31].

7.4 Data-Centric Optimizations

To cope with the exponentially growing amount of data, more sophisticated distributed storage systems and in-memory computing framework technologies might be adopted to minimize latency and resource usage.

7.4.1 Data Storage and Processing Optimization

Technologies like the Hadoop Distributed File System (HDFS) already help reduce energy costs by distributing data across multiple nodes to enable parallel processing [54]. This method can lower storage-related energy consumption by approximately 25% compared to older, centralized systems. Future enhancements to HDFS may incorporate intelligent block placement algorithms that account for real-time factors such as network congestion and node load, potentially surpassing current efficiency benchmarks.

Similarly, Apache Spark's in-memory computing framework minimizes latency while reducing power consumption in large-scale analytics workloads [38, 54]. Next-generation in-memory technologies are expected to feature adaptive caching and predictive data replication, improving how datasets are duplicated and migrated [38]. Advancements in data locality—optimizing the placement of computations near stored data—could further lower overhead by leveraging tiered storage systems that combine high-speed SSDs with lower-speed disks. As these techniques evolve, they can reduce energy costs and time-to-insight for data-driven operations [38].

7.5 Cloud-Native Architectures

Finally, as infrastructure increasingly shifts to cloud-native models, organizations will explore new approaches to virtualization, containerization, and microservices to gain flexible resource allocation that scales seamlessly with demand, all while curbing power consumption [38].

7.5.1 Integration of Cloud-Native Solutions

Cloud-native architectures are set to evolve alongside energy-aware orchestration systems capable of dynamically migrating workloads to the most power-efficient nodes, by abstracting hardware into virtualized or containerized layers, providers can rapidly adjust deployments to real-time resource demands, significantly reducing power consumption during peak usage periods [8]. Additionally, microservices and serverless paradigms, which divide applications into modular, on-demand components, offer further optimization by enabling each service to scale independently, avoiding the inefficiencies associated with monolithic architectures [8].

Multi-cloud and hybrid strategies will enhance these capabilities even further by dynamically distributing services across platforms or geographic regions in the future. This approach could leverage green energy sources, mitigate high-demand periods, and optimize energy usage on a global scale. Such advancements highlight the growing agility and decentralization of cloud computing, where resource management evolves into an ongoing, AI-driven process rather than a static architectural decision [54].

8 Future Strategies for Energy Efficiency in Machine Learning

As mentioned in the previous chapter, balancing growing performance demands with sustainability goals requires a comprehensive set of strategies, from physical infrastructure and renewable energy integration to intelligent resource management and data-centric optimizations. Below, we will introduce how cutting-edge hardware accelerators, AI-driven scheduling, model optimization, and sustainable infrastructure come together in a layered manner to shape the future of energy-conscious ML—reinforcing many of the innovations already underway in big data analytics.

8.1 Physical Infrastructure and Hardware Strategies

A strong foundation in data centre design and hardware acceleration can markedly reduce power usage for machine learning workloads. These physical-layer optimizations are consistent with the approaches discussed in the previous chapter, where efficient cooling and hardware placement contribute to substantial operational gains.

8.1.1 Advanced Hardware Strategies for Machine Learning Acceleration

As machine learning workloads become increasingly complex, specialized hardware accelerators are essential for achieving high performance and low energy consumption. GPUs, known for their unparalleled parallel processing capabilities, remain a cornerstone of cloud infrastructures but are responsible for nearly 50% of ML-related energy consumption in data centres [34]. To mitigate this, Tensor Processing Units (TPUs)—designed specifically for matrix and tensor computations—offer a more energy-efficient alternative. With approximately 30% lower energy consumption than GPUs while delivering comparable or superior throughput, TPUs have become a compelling choice for large-scale ML tasks [30].

Field Programmable Gate Arrays (FPGAs) present another avenue for reducing energy usage, particularly in ML inference, by leveraging targeted parallelism and reconfigurable logic. FPGAs have demonstrated significant energy savings over GPUs in practice [33]. However, their widespread adoption in large-scale cloud infrastructures has been hindered by the programming complexity and deployment overhead [33]. Future data centres may overcome these barriers by integrating toolchains and frameworks that simplify FPGA-based development, similar to emerging trends in big data analytics where user-friendly platforms are lowering the barrier to implementing specialized hardware solutions.

8.1.2 Data Centre Innovations Supporting ML Efficiency

Physical infrastructure improvements, such as liquid cooling systems and waste heat recovery, have significantly reduced power consumption, with waste heat reuse alone contributing to a 15% reduction in operational costs at certain facilities [13, 34]. As accelerators like TPUs and FPGAs are increasingly embedded in energy-optimized server racks, the power needed for intensive ML tasks is minimized [33]. Efforts to further refine cooling technologies—such as advanced liquid cooling loops adapted for high-density inference nodes—are expected to mirror the advancements seen in big data centres (e.g., free cooling and hot/cold aisle containment), illustrating a shared trajectory of infrastructure-level innovation that benefits both ML and big data analytics.

8.2 Green Energy Integration for ML

Similar to big data analytics, which utilizes renewable energy sources to minimize environmental impact, the shift toward greener AI ecosystems relies on comparable techniques and technologies, often powered by sophisticated AI-driven energy management algorithms.

8.2.1 Integrating Renewable Energy and Sustainable Infrastructure

Adopting solar and wind power in ML-focused data centres aligns with the industry-wide effort to lower carbon footprints. NVIDIA reports that AI algorithms capable of predicting energy demand and allocating workloads based on available green power have achieved energy savings of up to 30%, underscoring the synergy between operational efficiency and environmental responsibility [6]. Similarly, Microsoft Cloud employs AI-enabled energy forecasting to shut down unused resources, reducing idle power consumption dynamically [11]. These emerging strategies echo the green energy integration efforts in big data analytics, where intelligent load shifting across multiple regions or availability zones capitalizes on renewable generation. Over time, such integrated approaches are expected to enhance power usage effectiveness (PUE) for both ML-specific tasks and general data analytics pipelines, unifying the sustainability narrative across cloud computing domains [11].

8.3 Intelligent Resource Management

Moving beyond hardware and power sources, AI-powered dynamic scheduling offers a powerful avenue for optimizing resource use in machine learning.

8.3.1 Enhanced AI-Driven Dynamic Scheduling

AI-powered dynamic scheduling has become an essential tool for energy-efficient resource management in cloud computing. According to Ordonez et al., Dynamic Voltage and Frequency Scaling (DVFS) optimizes energy consumption during peak loads by dynamically adjusting CPU and GPU voltages in real-time, delivering up to 15% energy savings while maintaining system reliability [34]. When paired with machine learning-based workload prediction, DVFS can make fine-grained adjustments to prevent both under-provisioning and over-provisioning of computational resources [34].

In addition to DVFS, frameworks like Kubernetes are increasingly integrating AI and deep learning models for real-time task scheduling [50]. For example, reinforcement learning-based schedulers can more accurately predict usage patterns, enabling smarter resource allocation that reduces latency and lowers energy consumption by maximizing the utilization of available server nodes [61]. Solutions such as DynaSplit further enhance efficiency by splitting computations across heterogeneous platforms, dynamically offloading segments of neural network tasks to the most energy-efficient hardware [30]. These advancements reflect the intelligent orchestration strategies in big data analytics, where predictive analytics and cluster-level scheduling are employed to minimize resource waste and boost overall performance.

Observation (O-15): AI-driven dynamic scheduling (e.g., DVFS and Kubernetes-based RL) reduces energy consumption by 15–33% in ML workloads, mirroring optimization strategies in big data analytics and emphasizing cross-domain synergy.

8.4 Model-Centric Optimizations

Model compression technology and novel splitting strategies simplify calculations and are a technology trend with great potential for development. Their principles are similar to the data partitioning and memory processing mentioned in chapter 5.2.

8.4.1 AI-Specific Model Optimization Techniques

Reducing the energy footprint of machine learning models relies on several compression techniques, such as pruning, quantization, and knowledge distillation. According to Ordonez et al., these methods can lower inference-related energy consumption by up to 25%, especially for cloud-hosted models [34]. Another promising innovation is split computing, which distributes neural network layers between edge devices and cloud servers. This approach has demonstrated energy savings of up to 72% for large-scale models like Vision Transformer (ViT) while preserving high accuracy [30]. Such strategies parallel data partitioning in big data systems, reflecting the shared principle of distributing workloads to optimize resource utilization and minimize overall energy demands.

Layer	Innovations	Potential Impact
Physical Infrastructure	Liquid cooling, waste heat recovery, advanced layouts. For instance, Ericsson's waste heat recovery system and liquid cooling reducing server temperatures.	Reduces operational costs, decreases cooling energy, and improves energy efficiency.
Green Energy Integration	Solar and wind power, intelligent energy storage systems. Adoption of renewable energy, geographical load balancing, and advanced storage technologies.	Minimizes reliance on non-renewable energy, reduces carbon emissions, and ensures stable energy supply.
AI-Driven Resource Management	Dynamic workload balancing, predictive maintenance. AI optimizes resource allocation and equipment maintenance, e.g., OODA loop with K-means for cooling optimization.	Improves resource efficiency, lowers PUE ratings, reduces downtime, and minimizes energy waste.

Table 3: Future Advancements for Energy Efficiency

9 Conclusion

Observation (O-16): A multi-layered approach—combining hardware innovations, renewable energy integration, and model-centric optimizations—is essential for reconciling the growing computational demands of ML and big data with global sustainability goals.

Big data and machine learning have become indispensable to the success of cloud computing, yet their substantial energy requirements remain a pressing concern. This review has explored the energy profiles of these applications, assessed current optimization techniques, and charted potential paths toward greater energy efficiency. Among the key insights is the importance of adaptive, multi-faceted strategies combining advancements in algorithms, infrastructure enhancements, and integrating renewable energy sources. Future research should focus on converging intelligent resource management and sustainable design principles to address the dual goals of performance optimization and environmental stewardship. By leveraging these insights, cloud ecosystems can better meet the growing demands of data-intensive workloads while minimizing their ecological footprint.

References

[1] Abhishek Saini, Chaman Sharma, Nadeem Khan, Rohit Chauchan, and Gurjeet Singh. A REVIEW PAPER ON AWS. *EPRA International Journal of Multidisciplinary Research (IJMR)* (Jan. 2024), 164–169.

[2] Akhtar, T. Energy-efficient query processing strategies for distributed big data systems. *Power and Energy Journal 48*, 1 (March 2024), 1824–1839.

[3] Alarifi, A., Dubey, K., Amoon, M., Altameem, T., El-Samie, F. E. A., Altameem, A., Sharma, S. C., and Nasr, A. A. Energy-

Efficient Hybrid Framework for Green Cloud Computing. *IEEE Access 8* (2020), 115356–115369.

[4] AXBERG, T. Deriving an Natural Language Processing inference Cost Model with Greenhouse Gas Accounting: Towards a sustainable usage of Machine Learning.

[5] BADGUJAR, P. Optimizing ETL Processes for Large-Scale Data Warehouses. *Open Access 2*, 4 (2021).

[6] BLOG, N. Accelerated ai for energy efficiency, 2025. [Online].

[7] BORU, D., KLIAZOVICH, D., GRANELLI, F., BOUVRY, P., AND ZOMAYA, A. Energy-Efficient Data Replication in Cloud Computing Datacenters.

[8] BUYYA, R., AND GILL, S. S. Sustainable cloud computing: Foundations and future directions. *Cutter Business Technology & Digital Transformation Strategies 21*, 6 (Jun. 2018), 1–10. [Online].

[9] CHIHOUB, H.-E., IBRAHIM, S., LI, Y., ANTONIU, G., PEREZ, M. S., AND BOUGE, L. Exploring Energy-Consistency Trade-Offs in Cassandra Cloud Storage System. In *2015 27th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)* (Florianopolis, Brazil, Oct. 2015), IEEE, pp. 146–153.

[10] CHOI, D., JEON, H., LIM, J., BOK, K., AND YOO, J. Dynamic Task Scheduling Scheme for Processing Real-Time Stream Data in Storm Environments. *Applied Sciences 11*, 17 (Aug. 2021), 7942.

[11] CLOUD, M. Sustainable by design: Innovating for energy efficiency in ai, 2024. [Online].

[12] CUI, B., LI, Y., AND ZHANG, Z. Joint structured pruning and dense knowledge distillation for efficient transformer model compression. *Neurocomputing 458* (Oct. 2021), 56–69.

[13] EVENTS, N. The role of ai and machine learning in optimizing data centre energy use, 2025. [Online].

[14] FELLER, E., RAMAKRISHNAN, L., AND MORIN, C. Performance and energy efficiency of big data applications in cloud environments: A Hadoop case study. *Journal of Parallel and Distributed Computing 79-80* (May 2015), 80–89.

[15] GARCÍA-MARTÍN, E., RODRIGUES, C. F., RILEY, G., AND GRAHN, H. Estimation of energy consumption in machine learning. *Journal of Parallel and Distributed Computing 134* (Dec. 2019), 75–88.

[16] GOVERNMENT, S. P. P. Guiding opinions on promoting the high-quality development of the big data industry, 2021. Accessed: 2025-02-24.

[17] HONGKE ELECTRONICS TECHNOLOGY CO., L. Edge computing enables business continuity: A case of internet traffic surge. https://www.hongzdjs.com, 2020. Available online at www.hongzdjs.com.

[18] IM, J., LEE, J., LEE, S., AND KWON, H.-Y. Data pipeline for real-time energy consumption data management and prediction. *Frontiers in Big Data 7* (Mar. 2024), 1308236.

[19] JUAREZ, F., EJARQUE, J., AND BADIA, R. M. Dynamic energy-aware scheduling for parallel task-based application in cloud computing. *Future Generation Computer Systems 78* (2018), 257–271.

[20] KATAL, A., DAHIYA, S., AND CHOUDHURY, T. Energy efficiency in cloud computing data centers: a survey on software technologies. *Cluster Computing 26* (2023), 1845–1875.

[21] KAUSHIK, R. T., AND BHANDARKAR, M. GreenHDFS: Towards An Energy-Conserving, Storage-Efficient, Hybrid Hadoop Compute Cluster.

[22] KIM, Y. G., AND WU, C.-J. Autoscale: Energy efficiency optimization for stochastic edge inference using reinforcement learning. In *2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)* (2020), pp. 1082–1096.

[23] KOLLI, S. S. Evaluating Energy Consumption of Distributed Storage Systems : Comparative analysis.

[24] KUCHNIK, M., KLIMOVIC, A., ŠIMŠA, J., SMITH, V., AND AMVROSIADIS, G. Plumber: Diagnosing and Removing Performance Bottlenecks in Machine Learning Data Pipelines.

[25] KUM, S., OH, S., YEOM, J., AND MOON, J. Optimization of Edge Resources for Deep Learning Application with Batch and Model Management. *Sensors 22*, 17 (Sept. 2022), 6717.

[26] LERCHUNDI, A. G. Data analysis and machine learning approaches for time series pre-and post-processing pipelines. *University of The Basque Country* (2022).

[27] MAHARANA, K., MONDAL, S., AND NEMADE, B. A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings 3*, 1 (June 2022), 91–99.

[28] MARCULESCU, D., STAMOULIS, D., AND CAI, E. Hardware-aware machine learning: Modeling and optimization. In *2018 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)* (2018), pp. 1–8.

[29] MARKETS, AND MARKETS. Cloud computing market by service model (saas, paas, iaas), deployment model (public, private, hybrid), organization size, vertical (bfsi, retail, healthcare, manufacturing, it and telecom), and region - global forecast to 2025, 2025. Accessed: 2025-02-06.

[30] MAY, D., TUNDO, A., ILAGER, S., AND BRANDIC, I. DynaSplit: A Hardware-Software Co-Design Framework for Energy-Aware Inference on Edge, Oct. 2024. arXiv:2410.23881 [cs].

[31] MILIĆ, V. Next-generation data center energy management: a data-driven decision-making framework. *Frontiers in Energy Research 12* (Sept. 2024), 1449358.

[32] MUMUNI, A., AND MUMUNI, F. Data augmentation: A comprehensive survey of modern approaches. *Array 16* (Dec. 2022), 100258.

[33] NECHI, A., GROTH, L., MULHEM, S., MERCHANT, F., BUCHTY, R., AND BEREKOVIC, M. FPGA-based Deep Learning Inference Accelerators: Where Are We Standing? *ACM Transactions on Reconfigurable Technology and Systems 16*, 4 (Dec. 2023), 1–32.

[34] ORDONEZ, C., MACYNA, W., AND BELLATRECHE, L. Energy-Aware Analytics in the Cloud. In *International Workshop on Big Data in Emergent Distributed Environments* (Santiago AA Chile, June 2024), ACM, pp. 1–6.

[35] PARK, J., AND NO, A. Prune Your Model Before Distill It. In *Computer Vision – ECCV 2022*, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds., vol. 13671. Springer Nature Switzerland, Cham, 2022, pp. 120–136. Series Title: Lecture Notes in Computer Science.

[36] PATHANK, S. COMPARATIVE STUDY OF ENERGY USING APACHE SPARK AND HADOOP.

[37] PROKHORENKO, V., AND BABAR, M. A. Offloaded Data Processing Energy Efficiency Evaluation. *Informatica* (2024), 649–669.

[38] REALTY, D. Sustainable data centre ai, 2025. [Online].

[39] SAMSI, S., ZHAO, D., MCDONALD, J., LI, B., MICHALEAS, A., JONES, M., BERGERON, W., KEPNER, J., TIWARI, D., AND GADEPALLY, V. From Words to Watts: Benchmarking the Energy Costs of Large Language Model Inference, Oct. 2023. arXiv:2310.03003 [cs].

[40] SHANBHAG, S., AND CHIMALAKONDA, S. On the Energy Consumption of Different Dataframe Processing Libraries – An Exploratory Study, Sept. 2022. arXiv:2209.05258 [cs].

[41] SHANKAR, S., AND REUTHER, A. Trends in Energy Estimates for Computing in AI/Machine Learning Accelerators, Supercomputers, and Compute-Intensive Applications. In *2022 IEEE High Performance Extreme Computing Conference (HPEC)* (Waltham, MA, USA, Sept. 2022), IEEE, pp. 1–8.

[42] SHUVO, M. M. H., ISLAM, S. K., CHENG, J., AND MORSHED, B. I. Efficient acceleration of deep learning inference on resource-constrained edge devices: A review. *Proceedings of the IEEE 111*, 1 (2023), 42–91.

[43] SIDDIQA, A., KARIM, A., AND GANI, A. Big data storage technologies: a survey. *Frontiers of Information Technology & Electronic Engineering 18*, 8 (Aug. 2017), 1040–1070.

[44] SONG, X., SONG, A., XIAO, R., AND SUN, Y. One-step Spiking Transformer with a Linear Complexity. In *Proceedings of the Thirty-ThirdInternational Joint Conference on Artificial Intelligence* (Jeju, South Korea, Aug. 2024), International Joint Conferences on Artificial Intelligence Organization, pp. 3142–3150.

[45] SONI, D., AND KUMAR, N. Machine learning techniques in emerging cloud computing integrated paradigms: A survey and taxonomy. *Journal of Network and Computer Applications 205* (Sept. 2022), 103419.

[46] SOUALHIA, M., KHOMH, F., AND TAHAR, S. Task Scheduling in Big Data Platforms: A Systematic Literature Review. *Journal of Systems and Software 134* (Dec. 2017), 170–189.

[47] SOURAVLAS, S., ANASTASIADOU, S., AND KATSAVOUNIS, S. More on Pipelined Dynamic Scheduling of Big Data Streams. *Applied Sciences 11*, 1 (Dec. 2020), 61.

[48] SUN, B., KLODA, T., WU, C.-G., AND CACCAMO, M. Partitioned Scheduling and Parallelism Assignment for Real-Time DNN Inference Tasks on Multi-TPU. In *Proceedings of the 61st ACM/IEEE Design Automation Conference* (San Francisco CA USA, June 2024), ACM, pp. 1–6.

[49] TANG, D., DAI, M., SALIDO, M. A., AND GIRET, A. Energy-efficient dynamic scheduling for a flexible flow shop using an improved particle swarm optimization. *Computers in Industry 81* (Sept. 2016), 82–95.

[50] THINAKARAN, P., GUNASEKARAN, J. R., SHARMA, B., KANDEMIR, M. T., AND DAS, C. R. Kube-Knots: Resource Harvesting through Dynamic Container Orchestration in GPU-based Datacenters. In *2019 IEEE International Conference on Cluster Computing (CLUSTER)* (Albuquerque, NM, USA, Sept. 2019), IEEE, pp. 1–13.

[51] UCHECHUKWU, A., LI, K., AND SHEN, Y. Energy Consumption in Cloud Computing Data Centers.

[52] UNIVERSITY, C. Institute of big transportation data and artificial intelligence, 2020. Accessed: 2025-02-24.

[53] USMAN SANA, M., AND LI, Z. Efficiency aware scheduling techniques in cloud computing: a descriptive literature review. *PeerJ Computer Science 7* (May 2021), e509.

[54] VASHISHTH, T. K., SHARMA, V., PANDEY, A., AND TOMER, T. Innovative Advancements in Big Data Analytics: Navigating Future Trends With Hadoop Integration. In *Advances in Systems Analysis, Software Engineering, and High Performance Computing*, A. Kumari, Ed. IGI Global, May 2024, pp. 234–258.

[55] VISHWANATH, A., JALALI, F., HINTON, K., ALPCAN, T., AYRE, R. W. A., AND TUCKER, R. S. Energy Consumption Comparison of Interactive Cloud-Based and Local Applications. *IEEE Journal on Selected Areas in Communications 33*, 4 (Apr. 2015), 616–626.

[56] WANG, T., ZHOU, W., ZENG, Y., AND ZHANG, X. EfficientVLM: Fast and Accurate Vision-Language Models via Knowledge Distillation and Modal-adaptive Pruning, Oct. 2022. arXiv:2210.07795 [cs].

[57] WANG, X., ZHANG, R., SUN, Y., AND QI, J. KDGAN: Knowledge Distillation with Generative Adversarial Networks.

[58] WANG, Y., HAN, Y., WANG, C., SONG, S., TIAN, Q., AND HUANG, G. Computation-efficient deep learning for computer vision: A survey. *Cybernetics and Intelligence* (2024), 1–24.

[59] WU, C., BUYYA, R., AND RAMAMOHANARAO, K. Big Data Analytics = Machine Learning + Cloud Computing. In *Big Data*. Elsevier, 2016, pp. 3–38.

[60] WU, W. FIKIT: Priority-Based Real-time GPU Multi-tasking Scheduling with Kernel Identification, Feb. 2024. arXiv:2311.10359 [cs].

[61] XU, Z., GONG, Y., ZHOU, Y., BAO, Q., AND QIAN, W. Enhancing Kubernetes automated scheduling with deep learning and reinforcement techniques for large-scale cloud computing optimization. In *Ninth International Symposium on Advances in Electrical, Electronics, and Computer Engineering (ISAEECE 2024)* (Changchun, China, Oct. 2024), P. Siano and W. Zhao, Eds., SPIE, p. 175.

[62] YARALLY, T., CRUZ, L., FEITOSA, D., SALLOU, J., AND VAN DEURSEN, A. Batching for Green AI - An Exploratory Study on Inference. In *2023 49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)* (Durres, Albania, Sept. 2023), IEEE, pp. 112–119.

[63] YUAN, Y., SHI, J., ZHANG, Z., CHEN, K., ZHANG, J., STOICO, V., AND MALAVOLTA, I. The Impact of Knowledge Distillation on the Energy Consumption and Runtime Efficiency of NLP Models. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering - Software Engineering for AI* (Lisbon Portugal, Apr. 2024), ACM, pp. 129–133.

[64] ZHANG, Y., LIU, B., GONG, Y., HUANG, J., XU, J., AND WAN, W. Application of machine learning optimization in cloud computing resource scheduling and management. *Applied and Computational Engineering 64*, 1 (May 2024), 9–14.

[65] ZHOU, L., PAN, S., WANG, J., AND VASILAKOS, A. V. Machine learning on big data: Opportunities and challenges. *Neurocomputing 237* (May 2017), 350–361.

[66] ZÖLLER, M.-A., AND HUBER, M. F. Benchmark and Survey of Automated Machine Learning Frameworks. *Journal of Artificial Intelligence Research 70* (Jan. 2021), 409–472.

[67] ŠÍMA, J., VIDNEROVÁ, P., AND MRÁZEK, V. Energy Complexity of Convolutional Neural Networks. *Neural Computation 36*, 8 (July 2024), 1601–1625.