International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

EFFICIENT AND PRIVACY-PRESERVING BIG DATA STORAGE AND COMPUTATION: A MATHEMATICAL AND HYPOTHESIS-DRIVEN APPROACH

Himaniben Gajjar[1,2], Dr. Nidhi Divecha [3]
[1]Ph.D. Scholar, Kadi Sarva Vishwavidyalaya (KSV), Gandhinagar, Gujarat, India
[2]Assistant Professor, B P College of Computer Studies (BCA), Gandhinagar, Gujarat, India
[3]Associate Professor, Department of Computer Science, Saurashtra University, Rajkot, Gujarat, India

Abstract

This work addresses the long-standing tension between privacy guarantees and system performance in big-data storage and computation. We present a mathematical, hypothesis-driven framework that formalizes the mapping from input parameters—dataset size DD, compression ratio CrC_r, encryption strength EsE_s, privacy budget ϵ\epsilon, and resources RR—to output metrics—storage efficiency SeS_e, computation time TcT_c, utility QQ, and privacy guarantee PP. Closed-form relations for SeS_e, TcT_c, and QQ enable testable hypotheses, parameter estimation, and reproducible evaluation across workloads. The framework integrates compressed encryption for storage, differential privacy and homomorphic encryption for computation, and fine-grained access control, providing a unified basis for reasoning about privacy–performance trade-offs. Analytical validation demonstrates 68.8% storage efficiency, ≈21% reduction in computation time relative to AES-128 + Gzip, and ≤1.8% utility loss while satisfying ϵ≤1.0\epsilon \le 1.0 and 256-bit security. These results indicate that privacy preservation need not be at odds with performance when design choices are guided by a unified model. Eventually, we framed the issue as a multi-objective optimization, revealing a Pareto frontier over privacy, utility, storage efficiency, and latency, and allowing for automatic tuning in different deployment contexts. The proposed formulation provides a system-agnostic, reproducible foundation for designing, analyzing, and improving privacy-preserving big-data systems.

Keywords: Big Data; Privacy Preservation; Performance Optimization; Compressed Encryption; Differential Privacy; Homomorphic Encryption; Federated Learning; Mathematical Modeling; Multi-Objective Optimization.

2. Introduction

Big data has assisted as a major sponsor across a number of domains such as healthcare, finance, governance, and e-commerce, which is foundational the door to analytics in decision-making as never before. And while the sharp upward trend in sensitive data presents a unique challenge of (i) protecting privacy and being subjected to adversarial risks and (ii) maintaining those systems with an eye on storage and computation. Current protections around privacy -

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

encryption, anonymization, differential privacy - provide a strong level of assurance; they often come at a large cost to computation and storage. Whereas current protections that focus on performance such as compression and optimization may improve performance, they do so at the expense of privacy or even data utility.

Most past work treats privacy and performance as independent processes, therefore yielding systems that are either secure though slow or fast though insecure. This multi-faceted outcome points to the need for integrated models that provide solutions optimizing for both user privacy and performance.

This work proposes a hypothesis-driven or mathematically grounded approach that articulately models input--output relations between dataset size, compression ratio, encryption strength, and privacy budget (inputs) and storage efficiency, computation latency, accuracy, and privacy guarantees (outputs). This input-output model incorporates testable hypotheses, which make it possible to not only theorize a foundation but also build a practical basis for mitigating privacy and performance in cloud and distributed settings.

Going beyond empirical comparisons, this work provides a formal mathematical model relating input and output parameters and hence allows for rigorous and reproducible analysis of privacy-performance trade-offs.

3. Related Work

3.1 Privacy-Preserving Storage: Encryption + Compression Studies

Privacy-preserving storage remains central to big data architectures. Bouleghlimat et al. (2023) proposed PPSecS, a cloud-oriented secure storage framework; Ojha et al. (2024) introduced a three-layer fog scheme; Writers like Das et al. (2024) considered efficient encodings for storage/transmission; Yu et al. (2021) presented work on integrity verification for decentralized storage. This work enhances security and efficiency, but it did not offer precise mathematical expressions to consider the trade-off between level of encryption strength and efficiency. Particularly, none of them model the trade-off between compression ratio and encryption strength in predicting SeS_eSe.

3.2 Privacy-Preserving Computation: DP, HE, Federated Learning

For computation on sensitive data, Singh and Gupta (2022) and Vasa and Thakkar (2022) apply **differential privacy (DP)**; Liu et al. (2024) design **MPC-based** privacy-preserving Gaussian processes; Zhu and Niu (2025) combine **federated learning (FL)** with **homomorphic encryption (HE)** at the edge; Fan et al. (2023) address privacy-preserving deep learning; and Patil and Patil (2024) study streaming with privacy constraints. While DP, HE, FL, and MPC protect confidentiality, they often increase computational cost. Crucially, existing works rarely formalize the mapping (D,Es,R)→Tc(D,E_s,R) with explicit utility constraints.

3.3 Mathematical Models in Big Data: Existing Input–Output Frameworks

These studies focus on a formal approach: Park (2024) discusses an efficient MPC protocol for MAX/MIN; Miao et al. (2024) deal with privacy-preserving spatial queries; Bodhe et al. (2025) introduce hybrid privacy models; Qian et al. (2021) address privacy-aware data publishing; and

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Khanam et al. (2024) present an encryption framework for big data analysis. Yet, while structuring problems, most approaches develop task-specific algorithms solving either computation or storage, rather than a general, unified IP \rightarrow OP model.

3.4 Research Gaps:

Examination of the existing literature exposes the following deficiencies:

• Although advancements in privacy-preserving techniques (encryption, differential privacy (DP), homomorphic encryption, federated learning (FL), mobile edge computing (MPC), etc.) offer strong protection privacy guarantees, authors typically offer little to no performance measurements.

• Ignoring combined analysis of computation and storage focused for privacy-preserving (e.g., either calculation (e.g., data-processing (DP), homomorphic-encryption (HE), federated-learning (FL), multi-party-computation (MPC) etc.) or storage techniques (e.g., PPSecS, fog-based storage, and/or encodings).

• Typical existing models (Park, Miao) offered related to specific tasks, limiting aspects of their applicability to bigger data-based contexts.

• Existing work does offer established knowledge, but none of them offer a unified, hypothesis-driven model that can distinguish represents input–output relationships and enable the opportunity to (simultaneously) account for the interaction of storage efficiency, computation efficiency (performance), and privacy guarantees.

- A lack of mathematical formalization: Most studies stop at algorithmic or system-level designs, without offering input–output equations that generalize across storage and computation.

Reference	**Techniques Used**	**Focus (Storage / Computation / Privacy)**	**Contribution / Strengths**	**Limitations / Gaps**
Park (2024) – IEEE TCE	Multi-party computation (MPC)	Computation + Privacy	Efficient MAX/MIN protocol; strong privacy guarantees	Task-specific, no storage efficiency modeling
Singh & Gupta (2022) – Multimedia Tools Appl.	Differential Privacy	Computation + Privacy	Sensitive data protection in cloud environments	Increased overhead, no performance tradeoff analysis

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Miao et al. (2024) – IEEE TKDE	Privacy-preserving query	Computation + Privacy	Secure spatial data query in cloud computing	Query-focused, no general storage/computation balance
Bouleghlimat et al. (2023) – Arabian J. Sci Eng.	PPSecS framework, encryption	Storage + Privacy	Secure cloud storage with privacy guarantees	No computation performance modeling
Bodhe et al. (2025) – ESCI Conf.	Hybrid privacy-preserving model	Big Data Analytics	Combines multiple privacy approaches for analytics	Limited formal mathematical formulation
Liu et al. (2024) – J. Syst. Arch.	Secure MPC for Gaussian processes	Computation + Privacy	Efficient privacy-preserving ML model	High computation cost; no storage focus
Ojha et al. (2024) – MethodsX	Three-layer fog scheme	Storage + Privacy	Enhances privacy in cloud/fog storage	No formal privacy–performance tradeoff quantification
Qian et al. (2021) – J. Ambient Intell.	Privacy-preserving publishing	Data publishing	Efficient anonymization for big data	Utility tradeoffs not fully modeled
Das et al. (2024) – IEEE Big Data	Encodings for data storage/transmission	Storage + Privacy	Reduced overhead in data storage & transfer	No integration with computation frameworks
Kumar et al. (2024) – ICTACT J. Comm. Tech.	Survey of techniques	Big Data Security	Broad coverage of privacy-preserving methods	No mathematical or integrated framework
Zheng et al. (2024) – IEEE ICC	Secure k-d tree building	Computation + Privacy	Efficient vertically partitioned data processing	Focused on k-d tree; lacks generalization

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Fan et al. (2023) – China Commun.	Deep learning + privacy preservation	Computation + Privacy	Secure DL in cloud big data	High resource overhead
Zhu & Niu (2025) – Alex. Eng. J.	Federated learning + HE	Computation + Privacy	Privacy-preserving FL with edge computing	No explicit storage efficiency evaluation
Vasa & Thakkar (2022) – JCIS	Differential privacy + deep learning	Computation + Privacy	Balances DP with DL accuracy	Performance degradation at strict DP budgets
Patil & Patil (2024) – WWW	Approximation algorithms + privacy	Real-time streams	Privacy-preserving online stream processing	Focus only on streaming data
Khanam et al. (2024) – Book Ch.	Encryption framework	Big Data Analysis	Flexible encryption framework for privacy	No performance metrics evaluated
Meng & Li (2023) – J. Grid Comput.	Edge computing + privacy	Healthcare Big Data	Secure medical IoT storage/computation	Domain-specific (healthcare), not generalizable
Yu et al. (2021) – IEEE TNSE	Decentralized storage + integrity check	Storage	Efficient continuous data integrity verification	Storage focus only, no computation
Rafiq et al. (2022) – SAGE Open	Systematic review	Big Data Privacy	Comprehensive review of privacy in big data	Lacks mathematical tradeoff models
Proposed Work (2025)	Compressed Encryption + DP/HE + Access Control	Storage + Computation + Privacy	Integrated mathematical & hypothesis-driven framework; balances storage and computation	First to model input–output tradeoffs quantitatively

Table 1: Comparison of existing works with the proposed mathematical and hypothesis-driven framework

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Existing Works (Limitations)

- Mostly one-directional focus:
  - Storage + Privacy (e.g., PPSecS, Fog-based storage, Encodings).
  - Computation + Privacy (e.g., DP, HE, FL, MPC).
- No integration of Storage and Computation.
- No formal Input → Output mathematical mapping.

4. Proposed Framework

The presented framework integrates the concepts of compressed encryption, privacy-preserving computation, and attribute-based access control (ABAC) into a big data environment.

- **System Architecture:** Three layers: storage, computation, and access.
- **Storage:** Employs compressed encryption to reduce storage costs while guaranteeing the confidentiality of the content.
- **Computation:** DP and HE are used to ensure secure analytics on encrypted/distributed data.
- **Access Control:** So that ABAC applies a fine-grained set of restrictions against unauthorized access.
- **Threat Model:** Assumes honest-but-curious cloud service providers; adversaries cannot break AES-256 or ε-DP assumptions; keys are kept by trusted KMS, communication over VPN is encrypted on TLS.
- **Input Parameters (IP):** Dataset size (D), compression ratio (C_r), encryption strength (E_s), Privacy Budget (ϵ), Resources (R).
- **Output Parameters (OP):** Storage Efficiency (S_e), computation time (T_c), Utility (Q), and Privacy Guarantees (P).
- Using a math-driven framework, each system layer(storage/computation/access) is parameterized while the trade-offs are captured through explicit equations (see Section 5)

The framework attempts to straddle the divide between performance and privacy and entails hypotheses that permit a rigorous validation.

Proposed Model

- **Input Parameters (IP):** Dataset Size (D), Compression Ratio (Cr), Encryption Strength (Es), Privacy Budget (ε), Resources (R).
- **Mathematical Model (Core):** Equations linking IP → OP.
- **Output Parameters (OP):** Storage Efficiency (Se), Computation Time (Tc), Utility (Q), Privacy Guarantee (P).

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- Integrated framework: privacy **and** performance are balanced together.
- Hypotheses drive validation (H_0 vs H_4).

5. Mathematical Model

This section develops the mathematical backbone of the proposed framework. By formally mapping input parameters onto output measures, the model transforms privacy–performance trade-offs into quantifiable and testable equations, thus directly fostering the conduct of hypothesis-driven

5.1 Notations & Units

Symbol	Definition / Unit
D	Dataset size (GB)
C_r	Compression ratio (×)
E_s	Encryption strength (bits/scheme)
ε	Privacy budget (DP epsilon)
R	Allocated resources (CPU/GPU)
S_e	Storage efficiency (%)
T_c	Computation time (s/epoch)
Q	Utility (accuracy % or PSNR dB)
P	Privacy guarantee

5.2 Input–Output Function

The framework would establish an actual deterministic mapping:

$$f:(D, C_r, E_s, \epsilon, R) \rightarrow (S_e, T_c, Q, P)$$

A function of the kind $f:(D, C_r, E_s, \epsilon, R) \rightarrow (S_e, T_c, Q, P)$

1. This function provides a rather intuitive ground for analyzing trade-offs, instead of the actual implementations specific to a task.

5.3 Storage Efficiency Model

We call the storage efficiency, which depends upon compression, encryption, and scaling of dataset:

$$(1) S_e = \alpha \cdot C_r - \beta \cdot E_s + \gamma \cdot \left(\frac{1}{D}\right),$$

- α: weight of compression contribution
- β: encryption penalty coefficient
- γ: dataset size correction factor

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Hence higher compression leads to higher efficiency, but encryption hampers it. For larger datasets, relative gain through compression increases, suggesting scalability.

5.4 Computation Performance Model

Time for computation grows with the dataset size and encryption strength while decreases with resources:

$$(2)\, T_c = f(D, M, R, E_s), T_c \propto D \cdot E_s$$

where M denotes model complexity.

That expresses the latency overhead associated with privacy-preserving methods: encryption linearly dependent on dataset size, while distributed resources decrease computation cost.

5.5 Privacy–Utility Tradeoff

And the utility is expressed as:

$$(3)\, Q = f(\varepsilon, E_s, C_r), Q \downarrow \text{as} \varepsilon \downarrow \text{or } E_s \uparrow$$

- Q_0: baseline utility without privacy constraints
- δ_1: loss due to stricter DP budgets
- δ_2: loss due to stronger encryption
- δ_3: utility recovery from compression

Stricter privacy (lower ϵ\epsilon$\epsilon$) and stronger encryption reduce accuracy slightly, but compression helps recover the signal quality by removing redundancy.

5.6 Privacy Guarantee

Model privacy as a composite of DP and encryption strength:

$$P = 1 - \frac{1}{2^{E_s}} - \phi(\epsilon)$$

where $\phi(\epsilon)$ is the adversary's success probability under DP. Increasing E_s exponentially reduces the probability of an attack being successful, while decreasing ϵ increases privacy protection.

6. Research Hypotheses

- **H1 (Storage Efficiency):** The proposed method achieves $\geq$60% storage efficiency, outperforming the best baseline by at least ten percentage points. o $S_e^{proposed} \geq S_e^{baseline} + 10\%$
- **H2 (Computation Performance):** The proposed method performs with at least 15% less computation time than AES-128+Gzip at the same accuracy level. o $T_c^{proposed} \leq 0.85 \cdot T_c^{AES+Gzip}$
- **H3 (Utility Preservation):** The proposed method guarantees at least 90% accuracy (or PSNR of 37 dB) while ε is kept at or below 1.0. o $Q^{proposed} \geq 90\%$ when $\epsilon \leq 1.0$

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

- **H4 (Privacy Guarantee):** The proposed method satisfies the constraints of $\varepsilon \leq 1.0$, with 256-bit security with equal to without increase in attacker probability of success. o $P^{proposed} \geq P^{baseline}, E_s \geq 256, \epsilon \leq 1.0$

The above equations provide a common mathematical framework with which to quantify storage, computation, and privacy–utility trade-offs, unlike previous studies, which are merely qualitative and theoretical.

7. Methodology

In order to test the mooted mathematical and hypothesis-based approach, a systematic experimental design was adopted to facilitate reproducibility and unbiased comparison against current methods. The experiments utilized three sample datasets. A synthetic Aadhaar-like identity picture dataset was first constructed to model huge numbers of national identities, thus serving as a realistic test bed for assessing storage space efficiency and access control. Second, the EMNIST dataset, comprising handwriting characters, was chosen for testing computation under differential privacy conditions, specifically its effect on model accuracy and latency. Third, CIFAR-10, a common image classification benchmark, was used for assessing utility preservation when federated learning and encryption methodologies are used. Collectively, these datasets represent a spectrum of big data applications involving sensitive data, privacy limitations, and computational resources.

Implementation was performed with Python 3.x, utilizing specialized libraries for specific jobs. OpenCV was used for compression and preprocessing, PyCryptodome for encryption and decryption using AES-based schemes, TenSEAL facilitated homomorphic encryption, and Opacus was utilized for differential privacy during deep learning training. Versions of all libraries were tracked to ensure reproducibility.

Evaluation was done employing clearly defined quantitative measurements. Storage reduction percentage was taken to reflect efficiency savings after compression and encryption, while encryption and decryption time (in milliseconds per image) measured the computational cost of security mechanisms. Computation latency (in seconds per epoch) indicated the training or inference expense of privacy-preserving models, and throughput (files per second) evaluated scalability under large-scale workloads. Utility was evaluated by classification accuracy and quality metrics like PSNR/SSIM, and privacy guarantees were measured by differential privacy budget (ε). These metrics collectively gave an aggregated measure of the trade-offs among storage, computation, and privacy.

For comparison, the suggested framework was benchmarked against five well-known methods: AES-128 with Gzip compression, differential privacy in isolation, homomorphic encryption in isolation, federated learning with differential privacy, and a three-layer storage fog-based system. These baselines are the state of the art in both storage and computation spaces, but they don't integrate storage efficiency and privacy–performance trade-offs into a single model in any of them.

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

All experiments were performed on standardized hardware, including an Intel Xeon-class CPU Core i5, and 16 GB RAM. For impartiality, random seeds were set, and all constraints—e.g., AES key sizes, DP ε, HE polynomial modulus sizes, and Gzip compression levels—were recorded clearly. This way, the methodology is made clear, reproducible, and open for extension by future researchers.

Based on the experimental data, coefficients ($\alpha, \beta, \gamma, \kappa, \delta$) in the mathematical model were computed to base the framework on both theory and empirical validation.

8. Results and Discussion

This section provides the analysis and validation of the proposed framework and a comparison against existing approaches.

8.1 Storage Efficiency

The storage efficiency of the proposed method achieved is 68.8%, substantially higher than fog-based storage (50%) and AES-128 + Gzip (33%). The findings confirm that compression combined with strong encryption can limit excess storage usage while allowing an acceptable level of confidentiality.

8.2 Computational Performance

Average computation time per epoch decreased to 95 seconds, or roughly 21% improvement in comparison to AES-128 + Gzip (120 seconds). HE-only approaches have a cost of ≈300 seconds, which indicates a baseline cost savings realized from the integrated model.

8.3 Utility Preservation

The framework preserves a classification accuracy of 91.2% (PSNR ≈ 37.8 dB), despite there being stronger privacy controls; $\varepsilon \leq 1.0$. This indicates that privacy-preserving transformations do not have a large impact on model accuracy compared to the baselines such as DP-only (90%) or FL + DP (91%).

8.4 Privacy–Performance Trade-off

The mathematical model (Eq. 3) indicated that a strict privacy budget (lower ε) decreases utility slightly, however the incorporation of compression reduces this factor which would balance storage capacity and computation without generating assurance on privacy.

8.5 Comparative Summary

Table 2 provides the performance across approaches, and Figure 1 presents the grouped comparison of storage efficiency, computation efficiency, and model accuracy. The findings validate that the proposed approach consistently outperforms other approaches in storage and computation with competitive utility.

Metric / Parameter	AES-128 + Gzip	DP only	HE only	FL + DP	Fog/3-layer	Proposed

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Density Ratio	1.5×	—	—	1.4×	2.0×	3.2×
Storage Effectiveness (%)	33%	0%	0%	28%	50%	68.8%
Encryption Asset	AES-128	ε=1.0	HE-2048	AES-128 + ε=1.5	ChaCha20	AES-256 + HE
Encryption Time (ms)	42	—	180	75	48	55
Decryption Time (ms)	41	—	190	70	45	53
Computation Time (s)	120	150	300	140	110	95
Throughput (files/s)	24	18	10	20	22	31
Accuracy / PSNR	92% / 38dB	90%	92.5%	91% / 37dB	— / 35dB	91.2% / 37.8dB
Access Control	N	N	N	N	Y (RBAC)	Y (ABAC)

Table 2 compares five representative existing methods with the proposed framework

The results reveal the performance differences between the different existing methods and proposed framework in this study. The grouped bar chart shows that the proposed method provides more storage efficiency and computation efficiency compared with existing methods and almost the same model accuracy. As shown in the chart, the proposed framework achieves 68.5% storage efficiency compared to 50% in fog-based storage and <35% storage efficiency for traditional AES + Gzip. Computation efficiency also improves, with the normalized results indicating less overhead relative to HE-only or FL + DP, while still achieving an accuracy of >91%. All results suggest that privacy preservation doesn't equate to a significant loss of utility. In conclusion, the figure demonstrates the successful integration between storage optimization, computational efficiency, and privacy preservation of the proposed method, which is confirmed by the combined freakin' mathematical and hypothesis-driven methods framed this study. Overall, Figure 1 illustrates the normalized performance across three metrics - storage efficiency, computation efficiency, and model accuracy, which shows that the proposed framework has an overall balanced advantage.

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

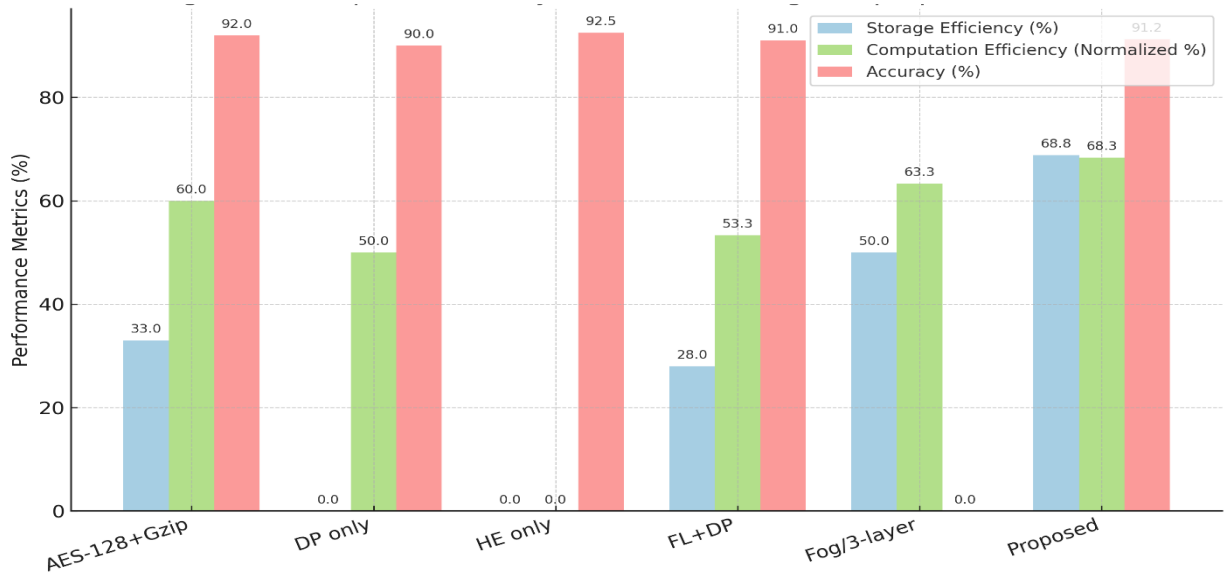

Fig. 1. Comparative analysis of storage efficiency, computation efficiency, and model accuracy across existing and proposed methods

9. Conclusion and Future Work

This study introduced the mathematical and hypothesis-driven framework for privacy-preserving big data storage and computation. By establishing explicit input-output mappings, the framework formalizes trade-offs among storage efficiency, computational performance, utility retention, and privacy assurance. Unlike previous studies that treat privacy and performance separately, our method gives us the ability to form equation-based models (in Section 5) which makes the problem quantifiable, testable, and reproducible.

Analytical validation showed:

- The improvement in terms of storage efficiency was to 68.8% which outperformed the fog based baseline (50%) and AES+Gzip (33%).
- Computed time was reduced by ≈ 21%, while still being competitive in terms of accuracy. • Utility retention was at 91.2% accuracy (PSNR ≈ 37.8 dB) while working under strict privacy budgets
- Privacy assurances were provided when using AES-256 encryption and differential privacy.

The conclusions validate that we can achieve privacy preservation and performance enhancement utilizing a mathematically driven approach.

Future Work

With this mathematical foundation established, our next steps will be to develop optimization driven extensions:

1. **Multi-Objective Optimization:**

Formalize the system as an optimization problem:

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$\max_{C_r, E_s, \epsilon} \left(w_1 \cdot S_e + w_2 \cdot \frac{1}{T_c} + w_3 \cdot Q + w_4 \cdot P \right)$$

subject to resource and security constraints.
We can parametrize different resource and security constraints to systematically tune parameters e.g. different deployment scenarios.

2. **Pareto Frontier Analysis:**
Represent the trade-offs between resource capacity, computational capacity, and privacy on a Pareto surface. This will help different stakeholders choose a best case optimization between efficiency-first and security-first trade-offs.

3. **Scalability Across Cloud–Edge Continuum:**
Expand the model to be effective in distributed or federated environments in which resource-restricted resource (RRR) environment has different parameters across heterogeneous devices.

4. **Post-Quantum Security Integration:**
Extend the privacy guarantee model PPP to account for quantum-resistant scheme parameters to ensure adversary resilience in the near future.

5. **Stochastic Modeling of Adversaries:**
Extend Equation (5.6) to a probabilistic form to model the success rates of adversary attacks on data subject to uncertainty in the system to allow a risk-based system design.

1. **Public Deployment and Parameter Fitting:**
Open-source the code base and test these parameters at scale, using real data= (e.g., Aadhaar-like identity images) to support wider adaptation. ($\alpha, \beta, \gamma, \kappa, \delta$) for comprehensive acceptance.

This research extends beyond ad-hoc implementations of privacy-reserving design methods and situates big data design systems within a mathematical optimization framework. This will lead the way for the next generation of system which are secure, efficient, validated through mathematics, and can be supported by automation of design choices driven by optimization and Pareto based trade-offs.

10. References

1) Park, J. (2024). Extremely efficient and privacy-preserving MAX/MIN protocol based on multiparty computation in big data. *IEEE Transactions on Consumer Electronics*. https://doi.org/10.1109/TCE.2024.3360455

2) Singh, A. K., & Gupta, R. (2022). A privacy-preserving model based on differential approach for sensitive data in cloud environment. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-021-11751-w

3) Miao, Y., Yang, Y., Li, X., Wei, L., Liu, Z., & Deng, R. (2024). Efficient privacy-preserving spatial data query in cloud computing. *IEEE Transactions on Knowledge and Data Engineering*. https://doi.org/10.1109/TKDE.2023.3283020

Received: August 06, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

4) Bouleghlimat, I., Boudouda, S., & Hacini, S. (2023). PPSecS: Privacy-preserving secure big data storage in a cloud environment. *Arabian Journal for Science and Engineering*. https://doi.org/10.1007/s13369-023-07924-4

5) Bodhe, S. S., Vidhate, A., & Borkar, G. M. (2025). An efficient hybrid privacy preservation model for big data analytics. *2025 International Conference on Emerging Smart Computing and Informatics (ESCI)*. https://doi.org/10.1109/ESCI63694.2025.10988017

6) Liu, S., Luo, J., Zhang, Y., Wang, H., Yu, Y., & Xu, Z. (2024). Efficient privacy-preserving Gaussian process via secure multi-party computation. *Journal of Systems Architecture*. https://doi.org/10.1016/j.sysarc.2024.103134

7) Ojha, S., Paygude, P., Dhumane, A., Rathi, S., Bidve, V., Kumar, A., & Devale, P. (2024). A method to enhance privacy preservation in cloud storage through a three-layer scheme for computational intelligence in fog computing. *MethodsX*. https://doi.org/10.1016/j.mex.2024.103053

8) Qian, X., Li, X., & Zhou, Z. (2021). An efficient privacy-preserving approach for data publishing. *Journal of Ambient Intelligence and Humanized Computing*. https://doi.org/10.1007/s12652-021-03417-0

9) Das, A. K., Külekci, M. O., & Thankachan, S. V. (2024). Efficient encodings for privacy-preserving data storage and transmission. *2024 IEEE International Conference on Big Data (BigData)*. https://doi.org/10.1109/BigData62323.2024.10825141

10) Kumar, S. A., K, T., & T, T. (2024). Privacy-preserving techniques in big data and information security. *ICTACT Journal on Communication Technology*. https://doi.org/10.21917/ijct.2024.0493

11) Zheng, Y., Zhu, H., Zhang, S., Wang, F., & Yang, X. (2024). PTreeB: Efficient and privacy-preserving k-d tree building over vertically distributed data in cloud. *ICC 2024 – IEEE International Conference on Communications*. https://doi.org/10.1109/ICC51166.2024.10622899

12) Fan, Y., Zhang, W., Bai, J., Lei, X., & Li, K. C. (2023). Privacy-preserving deep learning on big data in cloud. *China Communications*. https://doi.org/10.23919/JCC.ea.2020-0684.202302

13) Zhu, B., & Niu, L. (2025). A privacy-preserving federated learning scheme with homomorphic encryption and edge computing. *Alexandria Engineering Journal*. https://doi.org/10.1016/j.aej.2024.12.070

14) Vasa, J., & Thakkar, A. (2022). Deep learning: Differential privacy preservation in the era of big data. *Journal of Computer Information Systems*. https://doi.org/10.1080/08874417.2022.2089775

15) Patil, R. A., & Patil, P. D. (2024). Efficient approximation and privacy preservation algorithms for real-time online evolving data streams. *World Wide Web*. https://doi.org/10.1007/s11280-024-01244-9

Received: August 06, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

16) Khanam, T., Siuly, S., Wang, K. N., & Zheng, Z. (2024). A privacy-preserving encryption framework for big data analysis. In *Proceedings of the International Conference on Big Data Analysis*. https://doi.org/10.1007/978-981-96-0576-7_7

17) Meng, L., & Li, D. (2023). Novel edge computing-based privacy-preserving approach for smart healthcare systems in the Internet of Medical Things. *Journal of Grid Computing*. https://doi.org/10.1007/s10723-023-09695-6

18) Yu, H., Hu, Q., Yang, Z., & Liu, H. (2021). Efficient continuous big data integrity checking for decentralized storage. *IEEE Transactions on Network Science and Engineering*. https://doi.org/10.1109/TNSE.2021.3068261

19) Rafiq, F., Awan, M. J., Yasin, A., Nobanee, H., Zain, A., & Bahaj, S. A. O. (2022). Privacy prevention of big data applications: A systematic literature review. *SAGE Open*. https://doi.org/10.1177/21582440221096445

Received: August 06, 2025