# AN ADAPTIVE PRIVACY PRESERVING BASED ENSEMBLE LEARNING FRAMEWORK FOR LARGE DIMENSIONAL DATASETS

**CH. NANDA KRISHNA[1], K.F. BHARATI[2]**

[1]Research Scholar, JNTUA, Anantapuramu, Department of Computer Science and Engineering, India

[2]Associate Professor, JNTUACEA Anantapuramu, Department of Computer Science and Engineering, India

E-mail: [1]chnk1789@gmail.com, [2]kfbharathi@gmail.com

## ABSTRACT

With the rapid expansion of data, increasing computational power, and the complexity of high-dimensional datasets, it is of utmost importance to integrate a novel privacy-preserving model into deep learning frameworks. These frameworks, commonly utilized in traditional machine learning applications, heavily rely on extensive databases. Consequently, safeguarding the sensitive patterns generated by these approaches before they are uploaded to cloud storage becomes imperative. However, the development and implementation of a privacy-preserving deep learning model, specifically designed for highly dimensional cloud data, pose significant challenges. In cloud computing, traditional privacy-preserving deep learning frameworks prioritize data transformation methodologies over cryptographic approaches due to the substantial computational memory and time requirements involved. This preference is crucial to ensure privacy while simultaneously distributing multiple datasets in real-time multi-user applications. As the size of these applications expands, traditional privacy-preserving deep learning models require substantial computing resources to effectively preserve the intricate patterns generated by machine learning algorithms. To overcome these challenges, a unique privacy-preserving deep learning model is constructed, leveraging high-dimensional datasets and employing data partitioning techniques. Experimental findings validate the exceptional computational accuracy achieved by this model while simultaneously preserving privacy in the patterns, surpassing the performance of existing models.

**Keywords:** *Privacy Preserving, Machine Learning, Cryptography.*

## 1. INTRODUCTION

In today's rapidly evolving world, data is being generated at an unprecedented rate cross different fields. This has sparked a search for innovative methods to protect databases and optimize computational resources within organizations. Machine learning, a state-of-the-art technology that utilizes Artificial Intelligence, has emerged as a powerful tool for uncovering hidden patterns and insights from vast datasets. However, the sharing of electronic data for analysis poses a significant challenge to privacy[1]. This data often contains sensitive information, such as census, bank, or medical data, making it imperative to preserve the original data for accurate predictions. It is crucial to approach this issue with caution and avoid impractical solutions. Nevertheless, privacy can serve as a safeguard against the exposure of unwanted information during data mining projects aimed at achieving collective results. Privacy can be addressed at various levels in the field of data mining, and it is essential to implement measures that ensure both privacy and security for the entire database[2-4]. The collected datasets contain personal and sensitive information, and their release could potentially result in the unauthorized disclosure of confidential data. Ensuring the confidentiality of the data collection process is of utmost importance, as it contains sensitive information about individuals and organizations. These organizations gather and store this data for various purposes, and it is crucial to maintain its confidentiality and prevent any unauthorized disclosure. The application of machine learning models to this data has the potential to expose sensitive information, leading data owners to hesitate when considering publishing their data without privacy guarantees. However, relying solely on traditional solutions may not adequately address these concerns [5].

In the pursuit of data privacy, researchers have developed innovative methods to safeguard information and prevent unauthorized access. However, these methods present their own set of challenges. The models and algorithms designed to protect privacy often demand significant computational power and memory, making them unsuitable for selecting privacy patterns[6]. Therefore, it is imperative to explore alternative approaches to ensure privacy during data collection. One such approach involves storing data on a remote cloud server, serving as a secure repository for private information. This storage solution acts as a fortress, preventing external authorities from accessing the data and minimizing the risk of exposing sensitive information. Deep learning models play a crucial role in identifying patterns within scattered data. However, without privacy-preserving techniques, these patterns may inadvertently disclose confidential details. To address this concern, data owners exercise caution and only release their data when privacy is guaranteed. Conventional deep learning- based privacy-preserving models often fall short in selecting essential privacy patterns due to their demanding computational and memory requirements. Moreover, these models typically focus on a limited dataset from a single source, limiting their effectiveness. Fortunately, recent advancements in deep learning frameworks have revolutionized data processing techniques, particularly in the field of medical imaging. Feature extraction, classification, and segmentation have all greatly benefited from the deep layer-wise architecture and Convolution neural network, pushing the boundaries of data processing capabilities. To shield data from prying eyes, publishing techniques employ innovative methods such as transformation-based or cryptographic methodologies[7-10]. These methods cloak the data, ensuring its secure concealment and preventing unauthorized individuals from gaining access. Privacy-preserving procedures empower this versatile tool, rendering it applicable for diverse calculations. The astute modification of data presentation by users functions as a robust barrier, safeguarding the core attributes of the original data. The preservation of large datasets assumes paramount significance, given the potential for privacy breaches if data miners inappropriately process the information. To mitigate this peril, Privacy Preserving Data Publishing techniques are implemented to transform or encrypt the data before sharing it with trusted third parties. These sophisticated techniques guarantee the confidentiality and integrity of sensitive information [11]

When data is distributed across multiple servers or users, it is imperative to employ techniques that prioritize privacy and uphold confidentiality. In the first scenario, where data is dispersed among servers, each server independently divides and processes its specific portion of the data. In the second scenario, data is partitioned and processed locally on the users' devices. To ensure privacy, a range of techniques is utilized. Randomization methods, including scaling, rotation, and noise addition, modify the original data to eliminate any discernible patterns or connections. Moreover, cryptographic methods that employ encryption algorithms are implemented to safeguard the data against cyber threats. The Randomization algorithm is a widely-used technique in data mining to protect privacy.

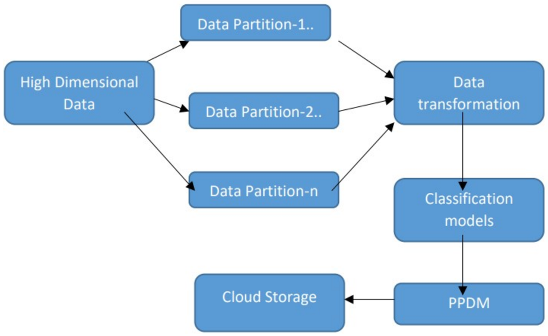


*Figure 1: Basic Privacy Preserving data mining framework on high dimensional data*

However, it is important to acknowledge that this approach may still have vulnerabilities when confronted with partially truthful adversaries. Nevertheless, randomization remains a valuable asset in centralized data mining systems, empowering users to input sensitive data without compromising their privacy. Figure 1 provides a visual representation of the traditional framework for maintaining privacy in classification learning with high- dimensional datasets. However, this framework has its limitations due to extensive computational memory and processing time requirements. To overcome these limitations, the data is divided into smaller subsets and trained using a partitioning technique. This partitioning technique enables accurate data prediction through the utilization of the PPDM model. To safeguard the privacy of predicted values, a differential privacy preserving method is employed. This method ensures that even if adversaries gain access to the predicted patterns, they are unable to extract sensitive information about the original data. The predicted patterns are securely stored on a cloud server. When analysing distributed data in partitioning models, two scenarios are taken into consideration. The first scenario involves the analysis of a substantial volume of data distributed across multiple servers. In this case, each server independently processes its designated portion of the data. The second scenario involves data being distributed among multiple users, with each user processing their specific portion of the data locally on their device. Dealing with a significant amount of data that requires substantial computing power presents a formidable challenge. However, there are situations where completing such a task becomes impractical. In such scenarios, it is advisable to divide the data into smaller segments, conduct distributed analysis, and consolidate the findings. Accurate predictions rely on a thorough examination of the data, which is where data partitioning and cluster analysis come into play. Specifically, when faced with a considerable volume of data that requires evaluation, the computational workload can become overwhelming, making it difficult to accomplish the task. In these cases, a more feasible alternative is to partition the data, perform distributed clustering, and subsequently merge the results. In contrast to a centralized database, where data is stored and managed in a single location, a distributed database operates on a different principle. In a distributed setup, data can be distributed vertically or horizontally across multiple sources. The drawback of a centralized database lies in the potential bottlenecks that may occur at critical points where data is released or assimilated due to the concentration of all the data in one place. As a result, accessing available data is less efficient compared to a distributed database system.

The primary contributions of this paper encompass two crucial aspects:

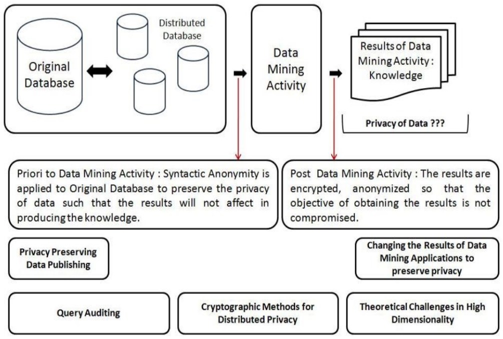
1) The implementation of multi-user data partitioning to expedite the data preparation process.

2) The adoption of a distributed database system to eliminate the need for a centralized database.

In order to overcome the challenges posed by noise or sparse values in the data partition, an advanced multi-layered strategy is employed for data pre-processing. This approach serves as the foundation for the development of a state-of-the-art ensemble deep learning framework, which not only ensures the preservation of privacy but also demonstrates remarkable proficiency when applied to the refined datasets. The subsequent sections of this thought-provoking essay are meticulously organized as follows: Section 2 critically examines the limitations associated with privacy-preserving models, Section 3 presents an innovative solution for safeguarding privacy in deep learning frameworks that handle high dimensional data, Section 4 provides a comprehensive analysis of the experiment's results, and finally, Section 5 concludes the work with future work.

## 2. RELATED WORK

Regulating the usage of data owned by others becomes a challenge when third parties access it. Although k-anonymity safeguards against identification, it doesn't always protect sensitive attributes from inference attacks. This is where the level of k-anonymity becomes significant. For example, a political candidate may use their opponent's medical history to influence public opinion. However, [11] have demonstrated that l-diversity isn't a reliable solution for attribute control, particularly when third parties are involved. The solution lies in effectively regulating data usage and preventing unauthorized access. When someone's identity is connected to information, it loses its anonymity. To protect individuals' privacy, data anonymization removes personal information from datasets. This ensures that data analysts and holders can share information without revealing people's identities. However, it can be difficult to publish data while maintaining privacy, as even small details can help identify individuals. This is why data anonymization is essential for safeguarding people's privacy. Once data is released, it's extremely challenging to retract it, which can result in negative consequences for those whose privacy has been violated [12]. Attribute linkage can result in privacy breaches. Although disclosure was initially created to overcome the limitations of k-anonymity, it also faces its own challenges. These challenges include the skewness attack, which happens when the data's distribution is uneven, and the similarity attack, which occurs when the equivalence class contains sensitive attribute values that have different but similar meanings. In the first scenario, l-diversity fails to prevent attribute disclosure because the distribution of the actual population differs from that of the dataset. Additionally, k-anonymity is only appropriate for datasets with a single row for each person, and it can cause excessive data distortion or privacy breaches in the case of a relational database. Bonchi recommends utilizing L-diversity to prevent attribute linkage attacks. L- diversity mandates that every quasi identifier (QID) class must possess at least one responsive attribute value, satisfying the k-anonymity requirement where k=l. Furthermore, L-diversity guarantees that each group includes a minimum of l sensitive attributes, rather than k members sharing the same QID. Nevertheless, L-diversity alone may not offer adequate protection against probabilistic attacks, as certain attributes are more prevalent than others[13-15]. Consequently, attackers can leverage probability theory to make assumptions about their target. Sensitive attributes possess considerable value due to their higher frequency of occurrence compared to other attributes. These attributes are isolated and regarded as anonymous, with isolation being a critical concept as a record is considered personal only if it cannot be detached from its surroundings. However, an adversary can take advantage of the situation by identifying data that has been extracted from the database, leading to a privacy breach. This violation is inherent when a server is anonymized. Furthermore, when all data is accessible as a single entity, attackers can effortlessly target the server without identifying any absent information. If data is chosen from the server by attackers, they must adjust their attack plan accordingly. Re-identifying personal information using quasi-identifiers [16] is one of the most common privacy breaches, which can be prevented by anonymization. [17] Suggests using k-anonymization as a technique, which aims to make each record similar to at least k-1 other records by suppressing or generalizing publicly available selected data. This technique enables sets of records with at least k members to connect sensitive data [18]. The values of the information attribute are widely recognized. Individuals in the original dataset can have their quasi-identifier values accurately predicted by a classifier trained on each projection. A set of minimal values, known as a "quasi-identifier," can be used with data from other sources to identify specific individuals. K-anonymity is a privacy protection method that preserves the original characteristics. However, traditional k-anonymity may not be suitable for census data. K- anonymity is the most widely used PPDM technique that maintains confidentiality. Ding et al. proposed a K-anonymity technique that divides the original dataset into data estimates, ensuring that each adheres to K-anonymity. To identify an unknown instance, one must use classification methods. Quasi-IDs can re-identify record owners by linking them to external sources. The evaluation of data loss depends on the data mining technique used to run the confidentiality algorithm, which is influenced by the context and affects the concept of secrecy. Personal data criteria and affiliation or classification restrictions vary in different scenarios. For example, if a health service provider shares patient data with third parties for research and analysis purposes, the privacy algorithm cannot disclose the sensitive attribute value. Maintaining data quality through a privacy protection strategy is crucial for enabling all possible data mining and analysis activities. Data privacy is maintained by keeping the end user from seeing the data's original features. In order to glean knowledge from the voluminous data sets amassed, data mining technologies operate on original data. The sensitive data sets include details about the privacy settings of specific people or any other secretive procedures. Such data will undoubtedly reveal sensitive information if processed directly by data miners, leading to a privacy breach. The data mining industry offers a variety of anonymity measures, which are typically described as data anonymization techniques, as illustrated in fig1. The algorithms for protecting privacy can be divided into two categories: distribution frameworks and techniques that amplify the original data source with random noise. Perturbation is a technique that has been utilized for a long time due to its effectiveness and simplicity. It works by replacing artificial data with real data that has similar statistical characteristics, making it difficult for attackers to extract sensitive information[19]. However, it is important to note that perturbation can only compute statistical parameters like minimum, maximum, average, and mean, which makes the data less valuable for humans.


Original Database
Distributed Database
Data Mining Activity
Results of Data Mining Activity : Knowledge
Privacy of Data ???
Privacy Preserving Data Publishing
Changing the Results of Data Mining Applications to preserve privacy
Query Auditing
Cryptographic Methods for Distributed Privacy
Theoretical Challenges in High Dimensionality

Priori to Data Mining Activity : Syntactic Anonymity is applied to Original Database to preserve the privacy of data such that the results will not affect in producing the knowledge.

Post Data Mining Activity : The results are encrypted, anonymized so that the objective of obtaining the results is not compromised.

*Figure 2: Traditional cryptographic based privacy preserving model*

To address this limitation, the additive noise perturbation method has been widely used to maintain the statistical features of the original data. The use of random noise can be a helpful tool in protecting sensitive information, but it is not always a completely effective solution. When additive noise is applied, there is a possibility that sensitive attributes may not be fully safeguarded. This is especially true when there is a high correlation between the QID and the sensitive attribute and the noise level is low. In such cases, the original value

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

of the sensitive attribute may still be identifiable even after applying perturbed data. Data swapping is another technique that can be used to help obscure sensitive information. This involves exchanging sensitive attribute values while ensuring that the swapping is only done within the same rank. However, this method also presents some risks. Even with data swapping, there is still a chance that sensitive information may be revealed if the correlation between the QID and the sensitive attribute is high. This is because the swapped value may still provide clues to the original sensitive attribute value, making it susceptible to inference. The process of synthetic data generation involves the creation of a statistical model from the original data. This is followed by the creation of a data table through sampling, after which fake data is used in place of real data. To ensure that the statistical characteristics of the synthetic data match those of the original data in each group, the data is divided into groups before synthetic data is generated for each group. Adjustments to the problem-solving context are necessary to calculate the expected return quantitatively using the perturbation function[20].

## 3. ADVANCED PRIVACY PRESERVING BASED DEEP LEARNING FRAMEWORK

In the pursuit of enhancing data processing efficiency and minimizing noise in complex datasets, a state-of-the-art approach has been developed.

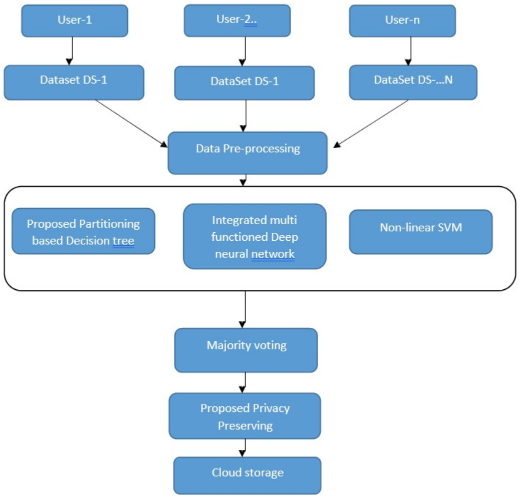


*Figure 3: Proposed Ensemble deep learning framework for Privacy preserving*

This approach involves a meticulous strategy for pre-processing and attributes splitting within a privacy-preserving framework that harnesses the power of ensemble deep learning. Each dataset undergoes a sophisticated transformation technique to eliminate any missing or sparse values that could impede comprehensive analysis on large-scale datasets. The resulting processed data is then fed into a unique ensemble deep learning model, meticulously designed to safeguard privacy. This model, illustrated in Figure 2, seamlessly integrates three classification techniques: deep multi-functional neural networks, non-linear support vector machines (SVM), and partitioning decision trees. To further fortify the security of deep learning patterns, an optimized integrity-based homomorphic privacy-preserving model is meticulously crafted. Ultimately, these sensitive privacy patterns are securely stored in remote cloud storage, ensuring both accessibility and the utmost level of confidentiality.

```
Input: Multi-source datasets MD = {D-1, D-2, ..., D-n}, Attribute A_tau, Max attribute value M_x, Minimum attribute value M_b, frequency of the maximum attribute value that contains class c Max_freq, frequency of the minimum attribute value that contains class c Min_freq.
Read input datasets MD
For each dataset D[i]
Do
    For each record I[r]
    Do
        For each attribute in I[r]
        Do
            If (A_tau[I] is Continuous && A_tau[I] = 0)
            then
                Replace A_tau[I] using the equation (1)
                A_tau[I] = (A_tau[I] - (M_b(A_tau) + M_x(A_tau)) / 2) / (Max_freq(A_tau) - Min_freq(A_tau))        (1)
            If (A_tau[I] is Categorical && A[I] == φ)
            then
                Replace A_tau[I] using the equation (2)
                A_tau[I] = (∑_{i=1, m=1}^{n, k} F(A_tau[i] / c_m) - F(c_m)) / (M_x * Prob(A_tau[i] / c_m))
                i = 1 to n; m = 1 to k (classes)        (2)
    Done
Done
```

**3.1 Input and Dataset Reading:** The algorithm starts by taking input, which includes multi-source datasets (MD), an attribute A_tau, maximum and minimum attribute values (M_x and M_b), and the frequencies of maximum and minimum attribute values containing a specific class (Max_freq and Min_freq). Then, it proceeds to read the input datasets

**3.2 Loop through Datasets:** The algorithm enters a loop to process each dataset in the input dataset set MD.

**3.3 Loop through Records:** Inside the dataset loop, the algorithm enters another loop to process each record in the current dataset D[i].

**3.4 Loop through Attributes:** Within the record loop, the algorithm iterates over each attribute in the current record I[r]. Continuous Attribute Check: It checks whether the attribute A_tau[I] is continuous and has a value of 0. If this condition is met, the algorithm proceeds to the next steps. If not, it skips to the next attribute. Continuous Attribute Transformation: For continuous attributes meeting the condition, the algorithm performs the transformation using Equation (1). This equation is used to normalize the attribute value based on its position between M_b(A_tau) (minimum attribute value) and M_x(A_tau) (maximum attribute value). The result is stored in A_tau[I]. Categorical Attribute Check: If the attribute A_tau[I] is categorical and A[I] is empty (φ), the algorithm proceeds to the next steps. If not, it skips to the next attribute. Categorical Attribute Transformation: For categorical attributes meeting the condition, the algorithm uses Equation (2) to transform the attribute value. This equation involves summation and probability calculations based on the frequency of attribute values for specific classes. The result is stored in A_tau[I]. Loop Closing: After processing all attributes in the current record, the algorithm proceeds to the next attribute loop iteration.Record Loop Closing: After processing all records in the current dataset, the algorithm proceeds to the next record loop iteration. Dataset Loop Closing: After processing all datasets in the input dataset set MD, the algorithm finishes and exits.

**Algorithm 2: Ensemble Deep learning framework**

Fast Random Forest:

For each attribute list in A_tau:

This loop iterates through each attribute list in the given set A_tau. If P-D[i] is Null:

This condition checks if a node P-D[i] is null, indicating the absence of data in a particular
node. Then:

If the node is indeed null, it proceeds to create a leaf node D, essentially ending the node's branch.

For each node n in N:

This loop iterates through each node in the set of nodes N.

If the count of instances in node n is equal to the total number of instances:

This condition checks if the count of instances in a particular node n is equal to the total number of instances.

Then:

If the condition is satisfied, it creates a leaf node n[i], indicating the end of this particular node's branch.

For each attribute A_tau:

This loop iterates through each attribute in the set A_tau.

Partition A_tau using different class labels using the RandomForest Partitioning Measure
(RPM):

This step involves calculating the RandomForest Partitioning Measure (RPM) for partitioning the attribute A_tau based on different class labels.

Choose the node with class m as the best split attribute in the partitioning list:

Among the partitioning options, the algorithm selects the node with class m as the best attribute to split the partitioning list.

Create a root node N[0]:

A root node N[0] is created to initiate the decision tree structure. Repeat the process until all nodes in the Tree T:

The algorithm repeats the above steps to create nodes and build the decision tree structure until all nodes in the Tree T are formed.

For each pattern in the Tree T:

This loop iterates through each pattern in the decision tree structure.

Apply Advanced Integrity Homomorphic Encryption (AIH) in the antecedent rule or consequent rule based on user choice:

Depending on user preferences, the algorithm applies Advanced Integrity Homomorphic Encryption (AIH) to either the antecedent or consequent rule of the pattern. Non-linear SVM:

Apply SVM multi-class optimization models using the given formulation:

This step involves applying a Support Vector Machine (SVM) model with multi-class optimization using the specified formulation.

The regularization term balances the model complexity and the loss:

The model includes a regularization term that helps balance the complexity of the model and the associated loss function.

The kernel function is defined for mapping input values to a higher-dimensional space:

A kernel function is utilized to map input values from their original space to a higher- dimensional space, allowing the algorithm to find more complex relationships.

Different cases for the kernel function based on input values are provided:

The kernel function has different cases based on the input values, determining how the mapping is done in different scenarios.

To each pattern in the decision tree construction, rule type is considered as either left side or right side of the pattern for privacy preserving. The advanced integrity based homomorphic encryption scheme used to preserve the left side or right side of the pattern is described below.

Third party Privacy Preserving Data Change Detection

Input parameters:

Ensemble classifier patterns

Pre-computed pattern integrity Pl

Group elements and keys (vm1, gm2, 01, 02, lambda1, PubkV) Cyclic group elements (T_alpha, T_beta, T_gamma)

Cyclic group generators (gm1, gm2) Transformation function (psi) Bilinear map (e)

Cloud server's randomized security parameter (k) Cloud parameters (CloudCParams)

Output: Data change Detection using integrity computation and signature. For each multi-user MC:

This loop iterates through each multi-user MC. Find the patterns that are related to MC:

This step involves identifying the patterns related to the current multi-user MC. For each pattern PMC[]:

This loop iterates through each pattern in PMC[].

Compute current Integrity using the ensemble learner patterns as IC(PMC[j]):

This step computes the current integrity using the ensemble learner patterns IC(PMC[j]). If (IP(PMC[j]) == IC(PMC[j])):

This condition checks if the initial pattern IP(PMC[j]) is equal to the computed pattern integrity IC(PMC[j]).

Then:

If the condition is met, the algorithm proceeds with the following calculations:

plaintext

Copy code

e(lambda2^psi, PubkV, gmtheta1^H(M_D)) = e(gmtheta2^(psi / (psi(m1 + lambda1))), gmtheta1^(m1) * gmtheta1^(lambda1))

e(gmtheta2^(m1 + lambda1), gmtheta1^(m1 + lambda1)) e(gmtheta2, gmtheta1)^((1 / (m1 + lambda1)) * (m1 + lambda1)) e(gmtheta2, gmtheta1)^1

SMC = e(gmtheta2, gmtheta1)

This calculation involves various group elements, keys, and operations to compute SMC. The provided pseudo-code is related to a privacy-preserving data change detection scheme using encryption and integrity computation. It involves various mathematical operations on group elements, keys, and parameters to ensure data privacy and detect changes while preserving confidentiality.

## 4. EXPERIMENTAL RESULTS

The experimental outcomes are emulated within the Amazon AWS cloud ecosystem. In this scenario, disparate medical datasets sourced from various origins are utilized as input. Subsequently, a privacy-preserving model is employed on the consolidated database. The proposed model incorporates a ground breaking homomorphic encryption technique centered on integrity preservation, which serves to safeguard the privacy of the ensemble deep learning model. Through experimentation, it has been demonstrated that the current model surpasses conventional methodologies in both privacy and accuracy.

*Table 1: Input Dataset And Its Features*

| Dataset Name | Acronym | Number of features | Number of data | Number of classes |
|---|---|---|---|---|
| Hepmass | HPDS | 28 | 3310786 | 2 |
| Statlog | SSDS | 10 | 43501 | 5 |
| Epileptic Seizure | ESDS | 179 | 11479 | 5 |
| Fried | FRDS | 11 | 40679 | 2 |
| Dataset Name | Acronym | Number of features | Number of data | Number of classes |

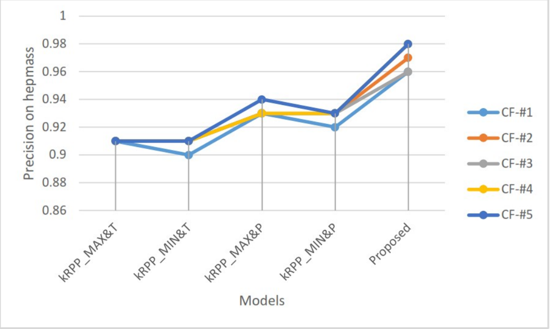


*Figure 5: Relationship between Precision and various k values*

For different values of k, sample decision tree patterns are simulated on the adult dataset. The decision patterns of k=5,25,45,65,85 are tabulated from table 1 to table 3 as shown below. Figure 4 presents the evaluation of various classification learning models using a perturbed hepmass dataset, incorporating different k values. The graph showcases the relationship between accuracy and the various k values

*Figure 4: Relationship between Accuracy and various k values*

Figure 5 presents the evaluation of various classification learning models using a perturbed hepmass dataset, incorporating different k values. The graph showcases the relationship between precision and the various k values. Figure 6 presents the evaluation of various classification learning models using a perturbed hepmass dataset, incorporating different k values. The graph showcases the relationship between AUC and the various k values.

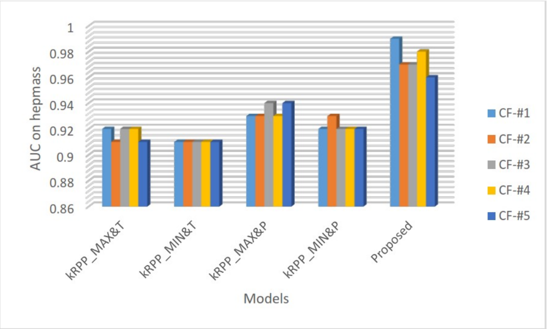


*Figure 6: Relationship between AUC and various k values*

The Table 1 represents the performance evaluation of a proposed K-anonymity decision tree pattern when K (the number of individuals a group must have in order to be considered anonymous) is set to 5. The evaluation is based on different metrics and compared against other methods

### 4.1 Test Samples (10%)

This indicates that the evaluation is performed on a subset of the dataset, specifically 10% of the total samples

### 4.2 kRPP_MAX&T

This is a method (likely a privacy-preserving technique) denoted by "kRPP_MAX" that aims to achieve K-anonymity. The "T" might stand for "tree." It's evaluated for its effectiveness in preserving privacy while allowing maximum tree growth.

### 4.3 kRPP_MIN&T

Similar to the previous method, “kRPP_MIN” also aims for K-anonymity, and "T" could refer to the tree. This method, however, aims for minimal tree growth while maintaining privacy.

### 4.4 kRPP_MAX&P

This method, denoted by “kRPP_MAX,” again probably for K- anonymity, is evaluated in terms of privacy preservation while allowing maximum pattern expansion.

### 4.5 kRPP_MIN&P

Similar to the previous method, “kRPP_MIN” is evaluated for privacy preservation, but this time with minimal pattern expansion growth.

### 4.6 Proposed

This column represents the proposed K-anonymity decision tree pattern, presumabl a novel approach. It is compared against the other methods in terms of its performance.

The numbers in the table represent performance scores, and it seems the scores are likely measures of how well each method achieves its goals. For example, higher values in the table might indicate better performance in terms of privacy preservation, tree growth, or pattern expansion, depending on the method being evaluated. The exact meaning of the numbers would depend on the context of the evaluation and the specific metrics used to assess each method's effectiveness.

## 5. CONCLUSION & FUTURE WORK

Utilizing the Java programming language, the experimental outcomes were acquired to evaluate the efficacy of a privacy-preserving model on both original and transformed datasets. Statistical metrics like accuracy, recall, precision, and runtime were computed to gauge the model’s capability to uphold privacy while retaining the fidelity of machine learning decision patterns. To ensure the preservation of privacy, adjustments were made to sensitive attributes. The ensuing experimental results were then juxtaposed among distinct privacy-preserving models, encompassing geometric perturbation, rotational perturbation, and PABIDOT. These models underwent testing on a variety of datasets, namely FRDS, WQDS, ELDS, and LRDS, all sourced from a specific research paper. The proposed algorithm was applied to the Bank Marketing dataset derived from the UCI repository. This dataset encompassed 17 attributes and 45,211 rows. Among these attributes, the attribute signifying whether the "client subscribed to term deposit" was singled out as sensitive. Notably, no identifying attributes needed to be eliminated from the provided dataset. The attributes of age, job, marital status, and education were regarded as quasi-identifiers. In this regard, age assumed the role of a numerical quasi-identifier, while job, marital status, and education were deemed categorical quasi-identifiers. Diverse utility measurements, encompassing loss, were employed to assess the effectiveness of generalized data.

In order to overcome the problems in the cloud computing environment, a hybrid integrity and security-based block-chain framework is designed and implemented on the large distributed databases. In this framework, a novel decision tree classifier is used along with non-linear mathematical hash algorithm and advanced attribute-based encryption models are used to improve the privacy of multiple users on the large cloud datasets.

## REFERENCES:

[1] M. Zhang, Z.-A. Li, and P. Zhang, "A secure and privacy-preserving word vector training scheme based on functional encryption with inner-product predicates," Computer Standards & Interfaces, vol. 86, p. 103734, Aug. 2023, doi: 10.1016/j.csi.2023.103734.

[2] M. Hiwale, R. Walambe, V. Potdar, and K. Kotecha, "A systematic review of privacy-preserving methods deployed with blockchain and federated learning for the telemedicine," Healthcare Analytics, vol. 3, p. 100192, Nov. 2023, doi: 10.1016/j.health.2023.100192.

[3] P. M. Sánchez Sánchez, L. Fernández Maimó, A. Huertas Celdrán, and G. Martínez Pérez, "AuthCODE: A privacy-preserving and multi-device continuous authentication architecture based on machine and deep learning," Computers & Security, vol. 103, p. 102168, Apr.2021, doi: 10.1016/j.cose.2020.102168.

[4] G. Shen, Z. Fu, Y. Gui, W. Susilo, and M. Zhang, "Efficient and privacy-preserving online diagnosis scheme based on federated learning in e-healthcare system," Information Sciences, vol. 647, p. 119261, Nov. 2023, doi: 10.1016/j.ins.2023.119261.

[5] V. Terziyan, D. Malyk, M. Golovianko, and V. Branytskyi, "Encryption and Generation of Images for Privacy-Preserving Machine Learning in Smart Manufacturing," Procedia Computer Science, vol. 217, pp. 91–101, Jan. 2023, doi: 10.1016/j.procs.2022.12.205.

[6] M. Field et al., "Infrastructure platform for privacy-preserving distributed machine learning development of computer-assisted theragnostics in cancer," Journal of Biomedical Informatics, vol. 134, p. 104181, Oct. 2022, doi: 10.1016/j.jbi.2022.104181.

[7] Z. Zhou, Q. Fu, Q. Wei, and Q. Li, "LEGO: A hybrid toolkit for efficient 2PC-based privacy-preserving machine learning," Computers & Security, vol. 120, p. 102782, Sep. 2022, doi:10.1016/j.cose.2022.102782.

[8] Y. Zhao, Y. Yu, Y. Li, G. Han, and X. Du, "Machine learning based privacy-preserving fair data trading in big data market," Information Sciences, vol. 478, pp. 449–460, Apr. 2019, doi: 10.1016/j.ins.2018.11.028.

[9] W. Briguglio, P. Moghaddam, W. A. Yousef, I. Traoré, and M. Mamun, "Machine learning in precision medicine to preserve privacy via encryption," Pattern Recognition Letters, vol.151, pp. 148–154, Nov. 2021, doi: 10.1016/j.patrec.2021.07.004.

[10] G. Deng, X. Duan, M. Tang, Y. Zhang, and Y. Huang, "Non-interactive and privacy-preserving neural network learning using functional encryption," Future Generation Computer Systems, vol. 145, pp. 454–465, Aug. 2023, doi: 10.1016/j.future.2023.03.036.

[11] B. Wang, H. Li, Y. Guo, and J. Wang, "PPFLHE: A privacy-preserving federated learning scheme with homomorphic encryption for healthcare data," Applied Soft Computing, vol. 146, p. 110677, Oct. 2023, doi: 10.1016/j.asoc.2023.110677.

[12] M. A. P. Chamikara, P. Bertok, I. Khalil, D. Liu, and S. Camtepe, "Privacy preserving distributed machine learning with federated learning," Computer Communications, vol. 171, pp. 112–125, Apr. 2021, doi: 10.1016/j.comcom.2021.02.014.

[13] S. Samet and A. Miri, "Privacy-preserving back-propagation and extreme learning machine algorithms," Data & Knowledge Engineering, vol. 79–80, pp. 40–61, Sep. 2012, doi: 10.1016/j.datak.2012.06.001.

[14] S. Mohseni, M. S. Pishvaee, and R. Dashti, "Privacy-preserving energy trading management in networked microgrids via data-driven robust optimization assisted by machine learning," Sustainable Energy, Grids and Networks, vol. 34, p. 101011, Jun. 2023, doi: 10.1016/j.segan.2023.101011.

[15] S. Sav, J.-P. Bossuat, J. R. Troncoso-Pastoriza, M. Claassen, and J.-P. Hubaux, "Privacy-preserving federated neural network learning for disease-associated cell classification," Patterns, vol. 3, no. 5, p. 100487, May 2022, doi: 10.1016/j.patter.2022.100487.

[16] S. Zapechnikov, "Privacy-Preserving Machine Learning as a Tool for Secure Personalized Information Services," Procedia Computer

Science, vol. 169, pp. 393–399, Jan. 2020, doi:10.1016/j.procs.2020.02.235.

[17] P. Li, T. Li, H. Ye, J. Li, X. Chen, and Y. Xiang, "Privacy-preserving machine learning with multiple data providers," Future Generation Computer Systems, vol. 87, pp. 341–350, Oct.2018, doi: 10.1016/j.future.2018.04.076.

[18] K. Edemacu, B. Jang, and J. W. Kim, "Reliability check via weight similarity in privacy- preserving multi-party machine learning," Information Sciences, vol. 574, pp. 51–65, Oct.2021, doi: 10.1016/j.ins.2021.05.071.

[19] R. Hamza and D. Minh-Son, "Research on privacy-preserving techniques in the era of the 5G applications," Virtual Reality & Intelligent Hardware, vol. 4, no. 3, pp. 210–222, Jun.2022, doi: 10.1016/j.vrih.2022.01.007.

[20] N. Nguyen, T. Connolly, J. Overcash, A. Hubbard, and T. Sudaria, "RWD103 Evaluating a Privacy Preserving Record Linkage (PPRL) Solution to Link De-Identified Patient Records in Rwd Using Default Matching Methods and Machine Learning Methods," Value in Health, vol.25, no. 7, Supplement, p. S595, Jul. 2022, doi: 10.