International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Hybrid Multi-Client Filter Based Feature Clustering and Privacy Preserving Classification Framework on High Dimensional Databases

Kavitha Guda*[1], Dr. K. Kavitha [2] & Dr. B. Sujatha [3]

Submitted: 11/10/2023 **Revised**: 29/11/2023 **Accepted**: 09/12/2023

Abstract: A multi-client perturbation based data clustering approach for privacy preserving multi-client data analysis is one of the best strategy for the multi-client data privacy applications. The approach is based on the concept of adding noise to the data, in order to make it difficult for an attacker to infer sensitive information about individual data points, while still allowing for meaningful analysis to be performed. The data is partitioned among multiple clients and each client applies a local clustering algorithm to their data. The clients then share their local clustering results with each other, but not the actual data. A global clustering is then constructed by combining the local clustering results.In addition to this, it proposes an optimal bayesian privacy preserving approach using advanced CP-ABE scheme. This approach uses the concept of ciphertext-policy attribute-based encryption (CP-ABE) to encrypt the data and to provide fine-grained access control to the data. The approach estimates the joint probability of the data across multiple clients and uses this estimation to calculate the Bayes Score, which is a measure of the accuracy of the classifier. By maximizing the Bayes Score, the method can select the optimal classifier for the multi-client data while preserving the privacy of the individual clients.Experimental results on different datasets demonstrate that the proposed approach achieves good clustering performance while preserving the privacy of the individual data points. The results also show that the proposed optimal bayesian privacy preserving approach using advanced CP-ABE scheme can effectively protect the privacy of the data while providing accurate results.

Keywords: *privacy preserving, multi-client privacy preserving model, ensemble classification and clustering.*

1.Introduction

Privacy-preserving machine learning (PPML) is an important area of research that aims to develop machine learning algorithms that can be applied to sensitive data without compromising the privacy of individuals. This is achieved by applying techniques such as differential privacy, homomorphic encryption, and secure multiparty computation to the data and the learning process. The privacy of the data subjects, or the people about whom the data is about and how it is used, is a serious concern in the current data-driven world. Differential privacy preserving machine learning (DP-ML) is a technique that allows for the protection of sensitive data while still allowing for machine learning algorithms to be applied to the data. It is a way of ensuring that the information that is learned from the data is not specific to any individual and cannot be used to re-identify or track individuals[1-2].

Secure multiparty computation is a method for performing complex computations on sensitive data without revealing the data to any of the parties involved. This is achieved by dividing the data into parts, performing the computation on each part separately and then combining the results to produce the final outcome. Researchers have also proposed various architectures for privacy-preserving machine learning such as using secure enclaves, isolated environments where sensitive data can be processed without exposure, and federated learning, where data is kept decentralized and machine learning algorithms run on the data in a distributed manner. However, there are challenges in balancing privacy and accuracy as well as scalability and data privacy in distributed systems. Additionally, trust and transparency in the use of the data, robust privacy-preserving machine learning algorithms and ethical and legal issues are also important considerations. The application of privacy-preserving machine learning in healthcare, in particular, has the potential to revolutionize the way medical data is collected, stored and analyzed, but also raises ethical and legal concerns.Secure multiparty computation (SMC) is a method for allowing multiple parties to jointly perform computations on sensitive data without revealing their individual data to each other. This is achieved by using techniques such as secret sharing, garbled circuits, and

*[1]*Research Scholar, Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar, Tamil Nadu*

kavitharddy.darshan@gmail.com

[2]*Associate Professor, Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar, Tamil Nadu*

[3]*Assistant Professor, Department of Computer Science and Engineering, University College of Engineering(a), Osmania University, Hyderabad*

secure function evaluation. One of the main advantages of SMC is that it allows for collaboration on sensitive data without compromising the privacy of individual parties.However, there are several challenges associated with SMC. One of the main challenges is computational efficiency. SMC algorithms are often computationally expensive and may not be practical for large-scale systems. This is because the computation has to be performed multiple times, once for each party, and the results have to be combined to produce the final outcome.Another challenge is communication overhead. In order to perform SMC, the parties need to communicate with each other, and this communication can be a bottleneck for large-scale systems. Additionally, SMC requires a high level of coordination and synchronization among the parties, which can be difficult to achieve in practice.Another important challenge is trust. In order for SMC to be effective, the parties need to trust each other and the SMC algorithm. This can be difficult to achieve in practice, especially in situations where the parties do not have a pre-existing relationship.Finally, SMC is not a one-size-fits-all solution and may not be suitable for all types of sensitive data and computations. For example, it may not be appropriate for data that is highly sensitive or for computations that require a high degree of accuracy.In summary, SMC is a powerful technique for privacy-preserving computation but it faces several challenges such as computational efficiency, communication overhead, trust and suitability. It is important to consider these challenges when choosing SMC as a solution and to carefully evaluate whether it is appropriate for the specific use case.Secure multiparty computation (SMC) is a method for allowing multiple parties to jointly perform computations on sensitive data without revealing their individual data to each other. The mathematical foundations of SMC are based on various techniques such as secret sharing, garbled circuits and secure function evaluation.One of the most basic and widely used techniques in SMC is secret sharing. Secret sharing is a method for distributing a secret among a group of parties such that certain subsets of parties can reconstruct the secret, while other subsets cannot. The mathematical foundations of secret sharing are based on algebraic structures such as finite fields and polynomials.Garbled circuits are another important technique in SMC. This technique allows parties to compute on encrypted data without the need to decrypt it first[3]. In summary, SMC is built upon various mathematical foundations such as algebraic structures, Boolean circuits, and secure computation protocols. These mathematical derivations allow for the creation of secure methods for parties to jointly perform computations on sensitive data without revealing their individual data to each other[4].

One of the ways to achieve privacy-preserving machine learning (PPML) is through the use of differential privacy (DP). DP is a mathematical framework that measures the privacy of a dataset and ensures that information learned from the data is not specific to any individual by adding noise to the data. The amount of noise added is determined by a privacy budget, which balances privacy and accuracy. Homomorphic encryption and secure multiparty computation (SMC) are also popular PPML methods, which enable computations on encrypted data and collaboration on sensitive data without revealing individual data, respectively. These approaches each have their own benefits and challenges and are becoming increasingly important as more sensitive data is collected and shared. Examples of application of PPML include healthcare, where patient information can be used for research, and finance, where multiple parties need to share information about fraudulent transactions.Additionally, PPML techniques can be used in various other fields such as genomics, natural language processing, and computer vision, where sensitive data is collected and shared. For example, in genomics, PPML can be used to protect patient's genetic information while still allowing for research and development of new treatments. In natural language processing, PPML can be used to protect sensitive information in text data such as emails, chat logs, and social media posts. In computer vision, PPML can be used to protect sensitive information in image and video data such as surveillance footage and medical images. Overall, PPML is an important tool that enables organizations and researchers to work with sensitive data while preserving the privacy of individuals[5-8].

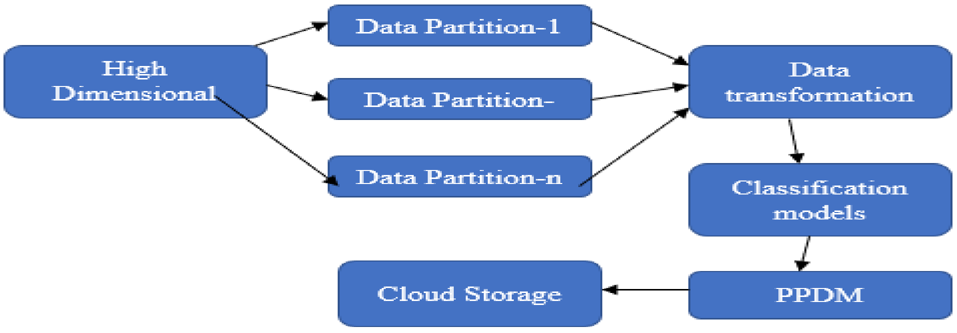


Fig 1: Basic Privacy Preserving data mining framework on high dimensional data

2.Related Works

The article covers different techniques that have been proposed and developed over the years to protect the privacy of individual data points while still allowing for accurate clustering and classification. These techniques include k-anonymity, l-diversity, t-closeness, Randomized response, Local Correlated Density Estimation etc.The article provides an overview of the different methods used in each technique, such as k-anonymity, which ensures that each data point has at least k-1 other similar data points in the same cluster, l-diversity, which guarantees that each cluster contains at least l different sensitive values, and t-closeness, which ensures that the distribution of sensitive values in each cluster is similar to the distribution in the whole dataset[9]. The article also covers Randomized response, which adds noise to the data to protect the privacy of the individuals, Local Correlated Density Estimation, which uses density estimation to group the data into clusters, Bayesian Ensemble Joint Distribution, which uses Bayesian models to estimate the joint distribution of multiple clusters and Modified Entropy-based Feature Ranking, which uses entropy to rank features and select the most relevant ones for clustering and classification.The article also discusses the challenges that have been addressed and need to be overcome in order to achieve effective cluster-based privacy preserving models. These challenges include scalability, communication overhead, maintaining the privacy-accuracy trade-off and dealing with new types of data and attacks. The article highlights the limitations of the existing techniques and suggests potential future research directions in this field.The authors also provide a summary of the advances made in the field of cluster-based privacy preserving models over the past 15 years and how they have evolved to provide better privacy guarantees while still maintaining the accuracy of the models. They also present a comparison of the different methods and techniques discussed in the article and their potential use cases. One of the most popular techniques for implementing differential privacy is the use of Laplace mechanism, which adds noise to the data based on the sensitivity of the query.[10] covers the different methods and techniques that are used for PPML in multi-client learning, such as differential privacy, homomorphic encryption, and secure multiparty computation. The article also discusses the challenges that need to be overcome in order to achieve privacy-preserving machine learning in multi-client learning, such as communication overhead and model convergence.[12] provides a survey of the challenges and solutions related to multi-client learning with non-IID (non-identically and independently distributed) data. The article covers the different methods and techniques that are used to handle non-IID data, such as data alignment, data augmentation, and domain adaptation. The article also discusses the challenges that need to be overcome in order to achieve effective multi-client learning with non-IID data, such as data heterogeneity, communication overhead, and model convergence. In conclusion, MPPDM is an active and important area of research that allows multiple parties to collaborate on a data mining task while keeping their data local. Multi-client learning, differential privacy, and secure multiparty computation are among the most popular approaches, each providing their unique benefits and challenges. There are a number of survey articles available on the topic of MPPDM that provide an in-depth overview of the field, including the different methods and techniques that are used, as well as the various challenges that need to be overcome in order to achieve privacy-preserving data mining in multi-client learning scenarios. Additionally, [15] provides a comprehensive survey of the field of PPML for multi-client learning with non-IID data. Non-IID data refers to data that is not identically and independently distributed among the parties involved. The article covers the different methods and techniques that are used for PPML in multi-client transfer learning, including data alignment, data augmentation, and domain adaptation. This article covers different techniques such as data perturbation, secure multiparty computation, and advanced CP-ABE schemes for protecting the privacy of individual data points while still allowing for accurate analysis. The article also discusses the challenges that need to be overcome in order to achieve effective multi-client PPML, such as scalability, communication overhead, and maintaining the privacy-accuracy trade-off.[16] covers different techniques such as joint probability estimation and Bayesian ensemble methods for protecting the privacy of individual data points while still allowing for accurate classification. The article also discusses the challenges that need to be overcome in order to achieve effective optimal bayesian privacy preserving classification, such as data heterogeneity, model convergence, and maintaining the privacy-accuracy trade-off. Multi-client privacy preserving machine learning is an active and important area of research that allows multiple parties to collaborate on a data mining task while keeping their data private. There are a number of survey articles available on the topic that provide an in-depth overview of the field, including the different methods and techniques that are used, as well as the various challenges that need to be overcome in order to achieve privacy-preserving data mining in multi-client scenarios. These include techniques such as data perturbation, secure multiparty computation, advanced CP-ABE schemes, optimal bayesian methods and many more. Additionally, many of these techniques have specific application in various fields such as Federated Learning with Non-IID Data, Federated Transfer Learning, Federated Learning for Time-Series Data and Federated Learning for Edge Computing[17-20].

3.Proposed Model

Data perturbation based multi-client privacy preserving clustering and classification is a method for allowing multiple parties to jointly perform clustering and classification on sensitive data without revealing their individual data to each other. The method is based on the concept of data perturbation, which is the process of adding noise to the data in order to protect the privacy of the individuals represented by the data.The basic idea of this method is that each party adds noise to their data before sharing it with the other parties. The parties then jointly perform clustering and classification on the perturbed data. The amount of noise added is determined by a privacy budget, which is a measure of the trade-off between the privacy of the data and the accuracy of the results.The data perturbation process can be done using various techniques such as adding random noise to the data, using randomized response techniques, or applying data transformation methods such as random projection.The clustering and classification process can be performed using various algorithms such as k-means, hierarchical clustering or decision trees.One of the main advantages of this method is that it allows for collaboration on sensitive data without compromising the privacy of individual parties. Additionally, it provides a way to control the trade-off between privacy and accuracy by adjusting the amount of noise added to the data.However, there are also some challenges associated with this method. One of the main challenges is the scalability of the method, as it may not be able to handle large amounts of data. Additionally, the method can be sensitive to the choice of data perturbation technique and clustering or classification algorithm used. In detail , data perturbation based multi-client privacy preserving clustering and classification is a method that allows multiple parties to jointly perform clustering and classification on sensitive data without revealing their individual data to each other by adding noise to the data in a controlled way. It provides a way to balance privacy and accuracy but also has scalability issues and sensitivity to the choice of techniques used as shown in figure 1.

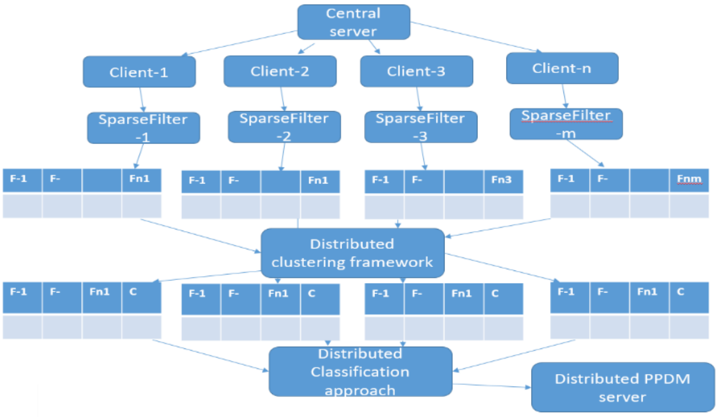


Fig 2: Overall framework of the proposed model

A sparse non-linear filter is a method used to identify and select a subset of the most informative and relevant features from a dataset. It uses a constraint that encourages parsimony in the number of features selected, it's a way of feature selection. Cluster-based classification, on the other hand, is a technique that groups data into different clusters based on their similarity, it's a way of data grouping and identifying patterns. These techniques can be applied in a vertical federated privacy preserving model where each client's data is processed independently. The sparse non-linear filter is used to extract relevant features from each client's dataset and cluster-based classification is used to group the data into different clusters. The model parameters are then sent back to each client to update their local data, and the process is repeated until the desired level of performance is reached. By using these techniques, a vertical federated privacy preserving model can effectively train machine learning models on multiple datasets while preserving the privacy of the individual data points. It allows for the use of a larger and more diverse dataset, which can lead to better model performance.

Optimal Gaussian Perturbation for sparse problem

Optimal Gaussian perturbation is a data transformation technique that is used to add noise to a dataset in order to protect the privacy of the individuals represented by the data. The noise is added in the form of random values that are sampled from a Gaussian distribution.The basic idea behind this technique is that by adding Gaussian noise to the data, it becomes more difficult to infer sensitive information about individual data points. The amount of noise added is determined by a privacy budget, which is a measure of the trade-off between the privacy of the data and the accuracy of the results.Gaussian perturbation is commonly used in privacy-preserving machine learning (PPML) in order to allow for the training of machine learning models on sensitive data without compromising the privacy of the individuals represented by the data. It can also be used in other contexts such as data publishing and data sharing. One of the main advantages of gaussian perturbation is that it is a relatively simple technique that can be easily integrated into existing machine learning pipelines. Additionally, it has been shown to be effective in preserving privacy while still allowing for accurate results in many PPML tasks. However, there are also some challenges associated with this technique. One of the main challenges is that the amount of noise added to the data needs to be carefully chosen in order to balance privacy and accuracy. Additionally, the technique is sensitive to the dimensionality and distribution of the data, and the choice of the privacy.

Non-linear gaussian estimation using numerical feature F is computed as:

$$\mathrm{NLG} = \sum_{j=0}^{N} \frac{1}{\sqrt{2\pi \log(F_j)}} \mathrm{Max}\{F_j\} / |\sum F_j|; i! = j\ -$$

Local co-related density estimation based clustering

Local correlated density estimation (LCDE) based clustering is a method for privacy-preserving clustering of multi-client data. In this method, each client independently applies a LCDE algorithm to their own data to estimate a local density function. The density function represents the probability of a data point belonging to a cluster and the LCDE algorithm estimates this function by taking into account the local correlation structure of the data.Once the local density functions are estimated, the clients share the density functions with each other but not the actual data. A global density function is then constructed by combining the local density functions. The global density function is then used to perform clustering on the multi-client data.One of the main advantages of LCDE based clustering is that it preserves the privacy of the individual data points as the clients only share the density functions, not the actual data. Additionally, LCDE based clustering can improve the performance of the clustering algorithm by taking into account the local correlation structure of the data.However, there are also some challenges associated with this method. One of the main challenges is that the LCDE algorithm can be sensitive to the choice of parameters, such as the bandwidth of the kernel function and the regularization term. Additionally, the method assumes that the local density functions are similar across clients, which may not always be the case.

Step 1: Each client partitions their data into PD.

Step 2: For each data point p[i] in PD,

Step 3: Apply weightage density to the data point using a Gaussian transformation function as defined in the formula provided.

$$wd_i = \sum_{j \in I, i!=j} exp(-(d_{ij}/d_c)^2)$$

$$d_{i,j} = \sqrt{\sum_{t=1}^{m} \left(x_i^t - x_j^t\right)^2}$$

Where d_c : Threshold

Step 4: Determine the highest density for each data point using the measures provided in the formula.

Step 5: Use the k-randomized centers as initial clusters and compute the k nearest neighbors for each center CPi.

The set of k nearest neighbors of center CPi is defined

$$\mathrm{NP}_i^k = \{P_j / \min(d_{ij}), i! = j\}$$

Step 6: Calculate the inter-cluster similarity and intra-cluster similarity for each k-neighbor initial cluster using the provided formula.

$$\lambda 1 = IntraClu(p_c, p_i) = \frac{1}{n_i - 1} hd_c. \sum_{m=1} d(p_c, p_m)$$

$$\lambda 2 = InterClu(p_c, p_i) = \min_{1<=m<=k} (\frac{1}{n_m} hd_c. \sum_{r=1} d(p_c, p_r))$$

$$\alpha = Q1;$$
$$\beta = Q2;$$
$$\chi = Q3;$$
$$UppOutlier = \chi + \Gamma\, max\{\lambda 1, \lambda 2\}. (\chi - \alpha)$$

$$LowerOutlier = \chi - \Gamma\, max\{\lambda 1, \lambda 2\}.(\chi - \alpha)$$

Step 7: Repeat the process until all points are assigned to k clusters or no more changes occur in the clusters.

Multi-client Feature ranking for classification

Multi-client privacy preserving modified entropy based feature ranking for classification is a method for selecting relevant features from a dataset while preserving the privacy of multiple clients.The basic idea of this method is to use a modified entropy based feature ranking algorithm that takes into account the privacy of multiple clients. Each client applies the algorithm to their own data and shares the ranked features with other clients, but not the actual data. A global ranking of the features is then constructed by combining the local rankings.The modified entropy based feature ranking algorithm is based on the concept of entropy, which is a measure of the disorder or uncertainty of a system. The algorithm calculates the entropy of each feature with respect to the class labels, and the features with the lowest entropy are considered to be the most informative and relevant.The privacy preserving aspect of this method is that clients only share the ranked features with each other and not the actual data. This allows for the selection of relevant features without compromising the privacy of individual clients.One of the main advantages of this method is that it allows for the selection of relevant features from multiple datasets while preserving the privacy of the individual clients. Additionally, it can improve the performance of the classification algorithm by selecting the most informative and relevant features.However, there are also some challenges associated with this method. One of the main challenges is that the modified entropy based feature ranking algorithm can be sensitive to the choice of parameters, such as the regularization term. Additionally, the method assumes that the ranked features are similar across clients, which may not always be the case.In this section, a hybrid feature ranking based decision tree model is constructed using the following equation as

$$Pr\,o\,posedFR = max\{\sqrt[3]{(\sum_{i=1}^{|D_i|}\sum_{j=1}^{|D_j|}(\sqrt[3]{D_i/|D_i|} - \sqrt[3]{D_j/|D_j|})^2), infoGain(D), Math.cbrt\big(E(D) * N * H\,D(data)\big)}$$
$$* E(D)/\big(\mathrm{E}(data)\big)\}$$

Where ,

D_i is the ith cluster class

D_j is the jth cluster class

Ensemble joint distribution based privacy preserving approach

A Bayesian Ensemble joint distribution based privacy preserving approach is a method for training machine learning models on sensitive data while preserving the privacy of the individuals represented by the data.The basic idea behind this approach is to use a Bayesian ensemble method to estimate a joint distribution of the data across multiple clients. The ensemble method combines the predictions of multiple models, each trained on a subset of the data, to produce a more accurate overall prediction.In this approach, the data is partitioned among multiple clients, and each client applies a local Bayesian model to their data. The clients then share their local models with each other but not the actual data. A global Bayesian ensemble model is then constructed by combining the local models. This ensemble model is used to make predictions on the multi-client data while preserving the privacy of the individual clients.One of the main advantages of this approach is that it allows for the training of machine learning models on sensitive data without compromising the privacy of the individuals represented by the data. Additionally, the Bayesian ensemble method can improve the performance of the overall model by combining the predictions of multiple models.However, there are also some challenges associated with this approach. One of the main challenges is that the Bayesian ensemble method can be computationally expensive, which may not be practical for large-scale systems. Additionally, the method can be sensitive to the choice of local Bayesian models used and the partitioning of the data among clients.

The proposed Bayesian score is computed as:

$$Pr\,o\,posed \text{ bayesian score estimation measure}$$
$$BayesJo\,intPr\,o\,b(D/s_i)$$
$$= \prod_{i=0}^{n}\prod_{j=0}^{q_i}\frac{\Gamma(\sum_{k=1}^{r}\alpha_{ijk}\,\chi(\alpha_{ijk}, \Gamma\, log(\,\alpha_{ijk}))))}{\Gamma(\sum_{k=1}^{r}\alpha_{ijk}\,\chi(\alpha_{ijk}, \Gamma\, log(\,\alpha_{ijk})) + \sum N_{ij})}\coprod_{k=1}^{r}\frac{\Gamma(\sum_{k=1}^{r}\alpha_{ijk}\,\chi(\alpha_{ijk}, \Gamma\, log(\,\alpha_{ijk})) + \sum N_{ijk})}{\Gamma(\sum_{k=1}^{r}\alpha_{ijk}\,\chi(\alpha_{ijk}, \Gamma\, log(\,\alpha_{ijk})))}$$

Class membership probabilities of each test samples are computed as

```
                    . $for$ each class in classlist
do
$To$ each attribute node $\text{AN}_i$ in BayesNet
 if($\text{AN}_i$==Class)
     Exponential-Log Class Probability =$\text{ECP}(\text{AN}_i)$=$\log\Gamma\sqrt[3]{ProbDist(\text{AN}_i)}$;
   else
    Exponential Attribute Probability =$\text{EAP}(\text{AN}_i) = \Gamma\sqrt{ProbDist(\text{AN}_i)}$;
$endfor$
```

Optimal Bayes Score with Joint Probability Estimations (OBS-JP) for privacy-preserving classification is a method for training machine learning models on sensitive data while preserving the privacy of the individuals represented by the data.The OBS-JP method is based on the concept of the Bayes Score, which is a measure of the accuracy of a classifier. The method estimates the joint probability of the data across multiple clients and uses this estimation to calculate the Bayes Score. By maximizing the Bayes Score, the method can select the optimal classifier for the multi-client data while preserving the privacy of the individual clients.In this method, the data is partitioned among multiple clients, and each client applies a local classifier to their data. The clients then share their local classifiers with each other, but not the actual data. The joint probability of the data is estimated based on the local classifiers and the Bayes Score is calculated. The optimal classifier is then selected by maximizing the Bayes Score.One of the main advantages of this method is that it allows for the training of machine learning models on sensitive data without compromising the privacy of the individuals represented by the data. Additionally, the OBS-JP method can improve the performance of the overall model by selecting the optimal classifier for the multi-client data.However, there are also some challenges associated with this approach. One of the main challenges is that the OBS-JP method can be computationally expensive, which may not be practical for large-scale systems. Additionally, the method is sensitive to the choice of local classifiers used and the partitioning of the data among clients.

Privacy on multi-client data

The mathematical steps for the setup, key generation, encryption, and decryption of multi-client CP-ABE include:

Setup:

Select a security parameter and randomly generate a pairing-friendly elliptic curve group G.

Select a random generator g in G.

Select a random number x and compute g^x. This is the master public key (MPK).

Select a random number y and compute g^y. This is the master secret key (MSK).

Key Generation:

Select a set of attributes A for the user.

Select a random number z and compute g^z. This is the user's public key (UPK).

Compute (g^x)^z * (g^y)^(Hash(A)) as the user's secret key (USK).

$Let\ H\_Attlist$ is the 512 value of user's integrity(MD5).

G1,G2 are the cyclic groups .A set of random generators from cyclic groups are r, g_r, g_p, r_j.

$$Cauchy\ \text{distribution}=\text{CD}(d)=\frac{b}{\pi[(d-a)^2+b^2]}$$

$$\text{SecrK.Dj} = \{\text{g\_r.mul}(H\_Attlist)\};$$

$$\text{SecrK.Dj}^* = \text{PK.gp.powZn}(r_j);$$

$$Secretkey = \{\text{CD(g\_r)},\text{SecrK.attr}, Attlist, S\ \text{ecrK.Dj},\text{SecrK.Dj}^*, H\_Attlist\}$$

Encryption:

Select a set of attributes A required to decrypt the message.

Select a random number r and compute g^r. This is the encryption key.

Encrypt the message m with the encryption key and the set of attributes A. The ciphertext is represented as (g^r, m*(g^r)^(Hash(A))).

Decryption:

Compute (g^r)^(Hash(A)) with the user's secret key.

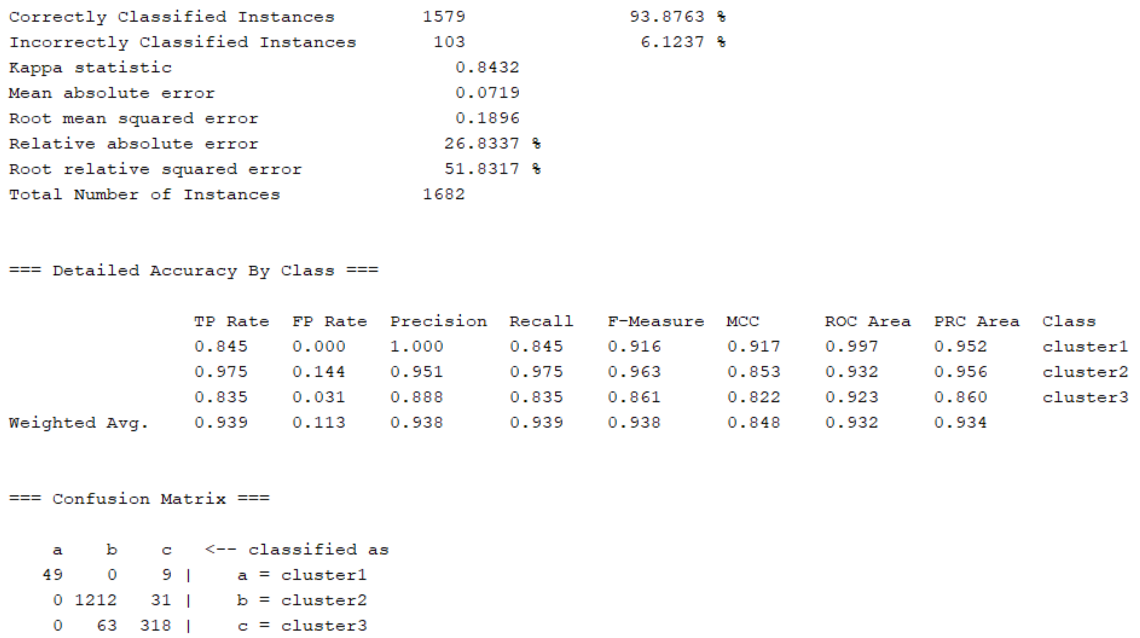
Divide the ciphertext by the computed value to obtain the original message.

Note: Hash(A) represents the hash value of the set of attributes A, and "^" represents exponentiation.

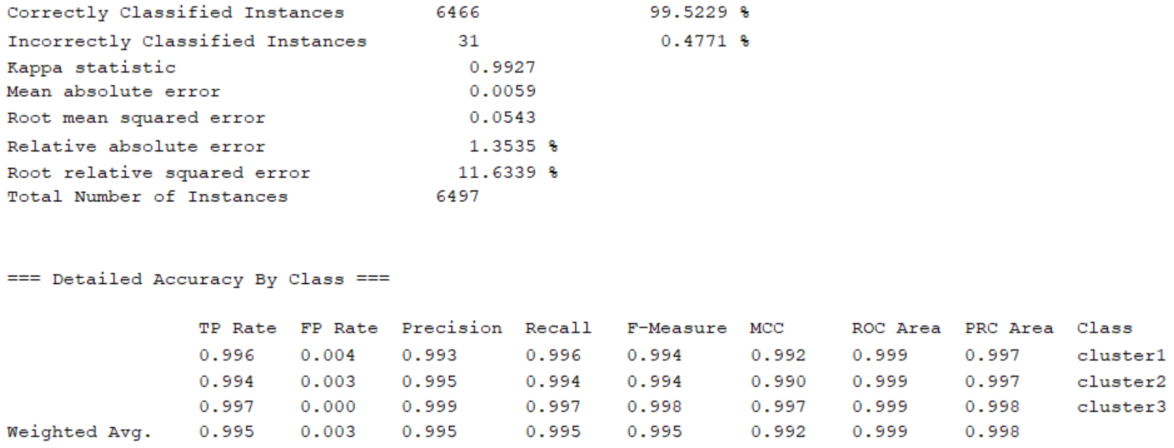
These steps provide a general overview of the mathematical operations involved in multi-client CP-ABE. The actual implementation and details may vary depending on the specific scheme used.

4.Experimental results

Experimental results on different datasets can include metrics such as accuracy, precision, and recall for the models trained on the protected data. When evaluating the performance of models trained on protected data, it is important to consider various metrics that can provide insight into the effectiveness of the privacy preserving technique.In additions to evaluating the performance of the models using these metrics, it is also important to compare the models trained on protected data with those trained on non-protected data. This comparison can demonstrate the effectiveness of the privacy preserving technique in maintaining the privacy of the individuals while still allowing for accurate analysis. Furthermore, in addition to evaluating the accuracy of the models, experimental results on UCI datasets can also include analysis of the privacy guarantees provided by the technique. This can include metrics such as the level of privacy protection, the level of data distortion, and the robustness of the technique against attacks. This will give a better understanding on how well the method can protect the privacy of the individuals represented by the data.



```
Correctly Classified Instances        1579               93.8763 %
Incorrectly Classified Instances       103                6.1237 %
Kappa statistic                          0.8432
Mean absolute error                      0.0719
Root mean squared error                  0.1896
Relative absolute error                 26.8337 %
Root relative squared error             51.8317 %
Total Number of Instances             1682


=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.845    0.000    1.000      0.845    0.916      0.917    0.997     0.952     cluster1
                 0.975    0.144    0.951      0.975    0.963      0.853    0.932     0.956     cluster2
                 0.835    0.031    0.888      0.835    0.861      0.822    0.923     0.860     cluster3
Weighted Avg.    0.939    0.113    0.938      0.939    0.938      0.848    0.932     0.934


=== Confusion Matrix ===

    a    b    c   <-- classified as
   49    0    9 |    a = cluster1
    0 1212   31 |    b = cluster2
    0   63  318 |    c = cluster3
```



```
Correctly Classified Instances        6466               99.5229 %
Incorrectly Classified Instances        31                0.4771 %
Kappa statistic                          0.9927
Mean absolute error                      0.0059
Root mean squared error                  0.0543
Relative absolute error                  1.3535 %
Root relative squared error             11.6339 %
Total Number of Instances             6497


=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.996    0.004    0.993      0.996    0.994      0.992    0.999     0.997     cluster1
                 0.994    0.003    0.995      0.994    0.994      0.990    0.999     0.997     cluster2
                 0.997    0.000    0.999      0.997    0.998      0.997    0.999     0.998     cluster3
Weighted Avg.    0.995    0.003    0.995      0.995    0.995      0.992    0.999     0.998
```

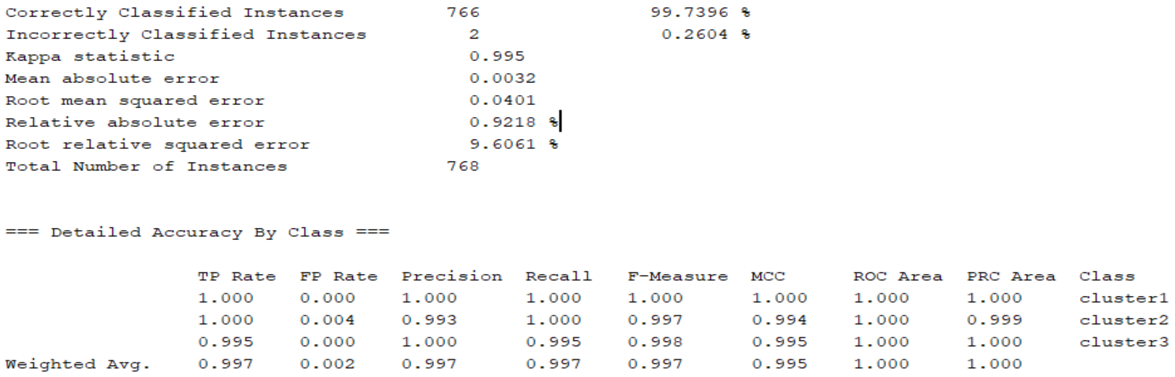
Table 1: Proposed cluster results and pattern on diabetes dataset

Cluster assign 4,78,0.236 label 1
Cluster assign 1,78,0.496 label 1
Cluster assign 0,84,0.433 label 1
Cluster assign 2,88,0.326 label 1

```
Cluster assign 2,52,0.141  label 1
Cluster assign 3,78,0.323  label 1
Cluster assign 8,86,0.259  label 2
Cluster assign 2,88,0.646  label 1
Cluster assign 2,56,0.426  label 1
Cluster assign 2,75,0.56  label 1
Cluster assign 4,60,0.284  label 1
Cluster assign 0,86,0.515  label 1
Cluster assign 8,72,0.6  label 2
Cluster assign 2,60,0.453  label 1
Cluster assign 1,74,0.293  label 1
Cluster assign 11,80,0.785  label 2
Cluster assign 3,44,0.4  label 1
Cluster assign 1,58,0.219  label 1
Cluster assign 9,94,0.734  label 2
Cluster assign 13,88,1.174  label 2
Cluster assign 12,84,0.488  label 2
Cluster assign 1,94,0.358  label 1
Cluster assign 1,74,1.096  label 1
Cluster assign 3,70,0.408  label 1
Cluster assign 6,62,0.178  label 2
Cluster assign 4,70,1.182  label 1
Cluster assign 1,78,0.261  label 1
Cluster assign 3,62,0.223  label 1
Cluster assign 0,88,0.222  label 1
Cluster assign 8,78,0.443  label 2
Cluster assign 1,88,1.057  label 1
Cluster assign 7,90,0.391  label 2
Cluster assign 0,72,0.258  label 1
Cluster assign 1,76,0.197  label 1
Cluster assign 6,92,0.278  label 2
Cluster assign 2,58,0.766  label 1
Cluster assign 9,74,0.403  label 2
Cluster assign 9,62,0.142  label 2
Cluster assign 10,76,0.171  label 2
Cluster assign 2,70,0.34  label 1
Cluster assign 8,78,0.516  label 2
Cluster assign 0,64,0.51  label 1
Cluster assign 3,44,0.14  label 1


Options: -C 0.25 -M 2
```

Ensemble DT Patterns

```
preg <= 4
|  pres <= 0: cluster1 (24.0)
|  pres > 0
|  |  preg <= 3: cluster3 (403.0)
|  |  preg > 3
|  |  |  pres <= 72: cluster3 (38.0)
|  |  |  pres > 72: cluster2 (27.0/2.0)
preg > 4
|  pres <= 0: cluster1 (11.0)
|  pres > 0: cluster2 (265.0)
```



```
Correctly Classified Instances         766               99.7396 %
Incorrectly Classified Instances         2                0.2604 %
Kappa statistic                          0.995
Mean absolute error                      0.0032
Root mean squared error                  0.0401
Relative absolute error                  0.9218 %
Root relative squared error              9.6061 %
Total Number of Instances              768

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     cluster1
                 1.000    0.004    0.993      1.000    0.997      0.994    1.000     0.999     cluster2
                 0.995    0.000    1.000      0.995    0.998      0.995    1.000     1.000     cluster3
Weighted Avg.    0.997    0.002    0.997      0.997    0.997      0.995    1.000     1.000
```

Note :

Filter+Kmeans, Filter+DBScan, Fitler+FeatureClustering : Here Filter represents the extreme outlier detection approach.

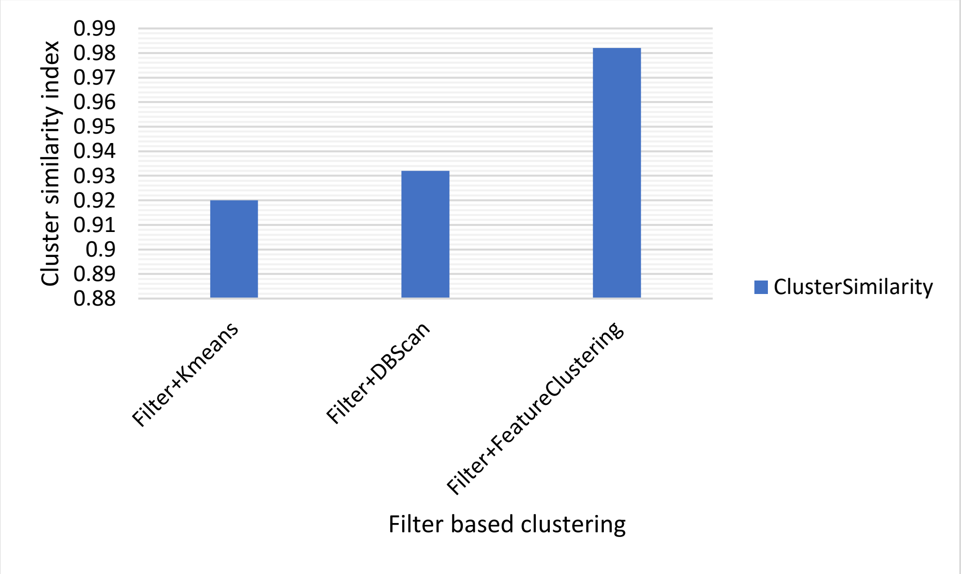


Fig 2: Comparative analysis of proposed approach to the conventional approaches for normalized mutual information based similarity index.

Fig 2, represents the comparative analysis of proposed approach to the conventional approaches using the contextual normalized mutual information index. This measure is used to find the best quality clusters than the conventional approaches using NMI measure.

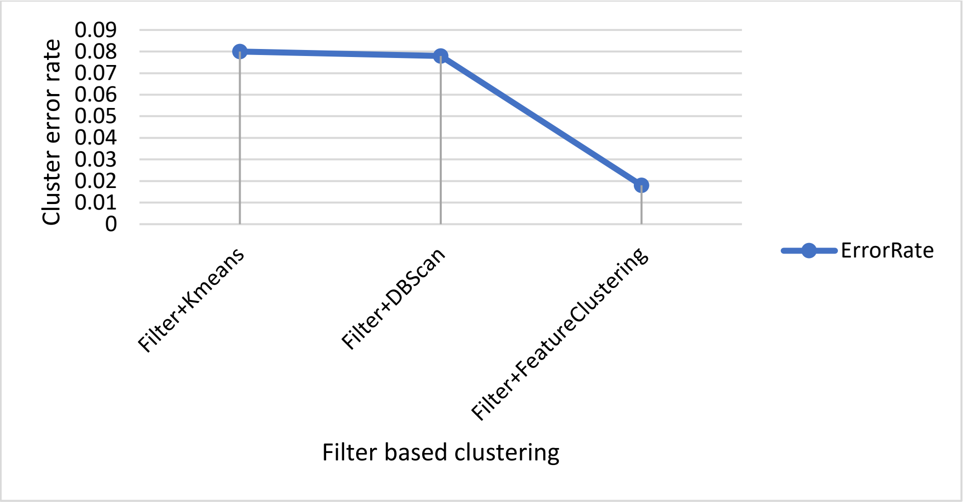
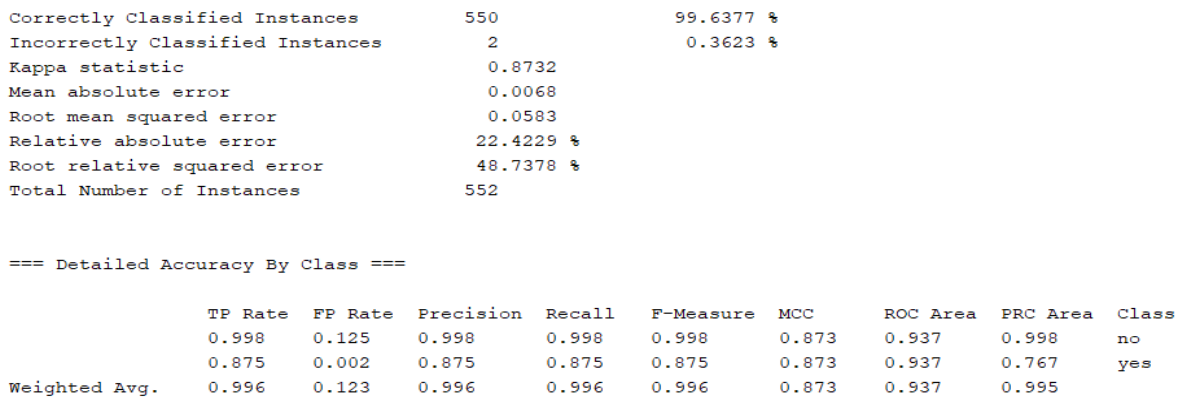


Fig 2: Comparative analysis of proposed approach to the conventional approaches for mean error rate for the clusters.

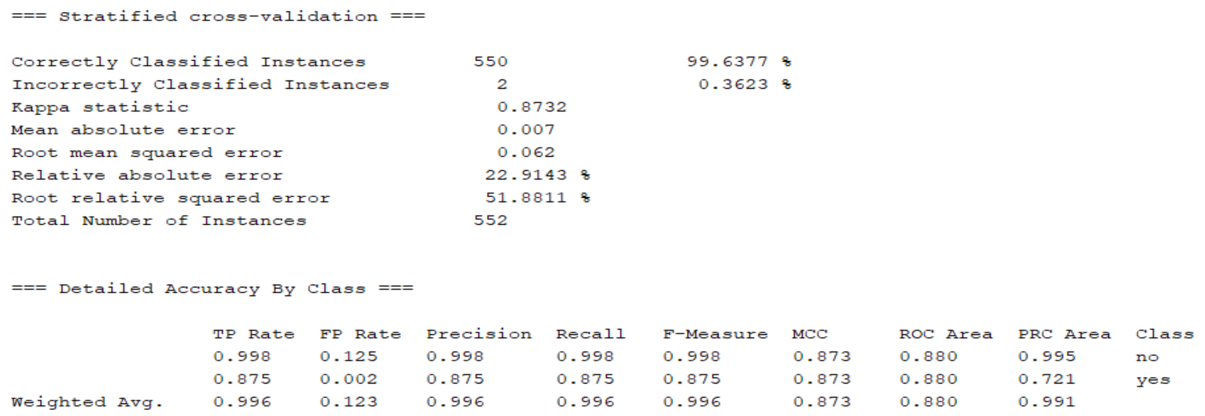
Fig 2, represents the comparative analysis of proposed approach to the conventional approaches using the contextual cluster error rate. This measure is used to find the best quality clusters than the conventional approaches using mean error rate.



```
Correctly Classified Instances         550               99.6377 %
Incorrectly Classified Instances         2                0.3623 %
Kappa statistic                          0.8732
Mean absolute error                      0.0068
Root mean squared error                  0.0583
Relative absolute error                 22.4229 %
Root relative squared error             48.7378 %
Total Number of Instances              552

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.998    0.125    0.998      0.998    0.998      0.873    0.937     0.998     no
                 0.875    0.002    0.875      0.875    0.875      0.873    0.937     0.767     yes
Weighted Avg.    0.996    0.123    0.996      0.996    0.996      0.873    0.937     0.995
```

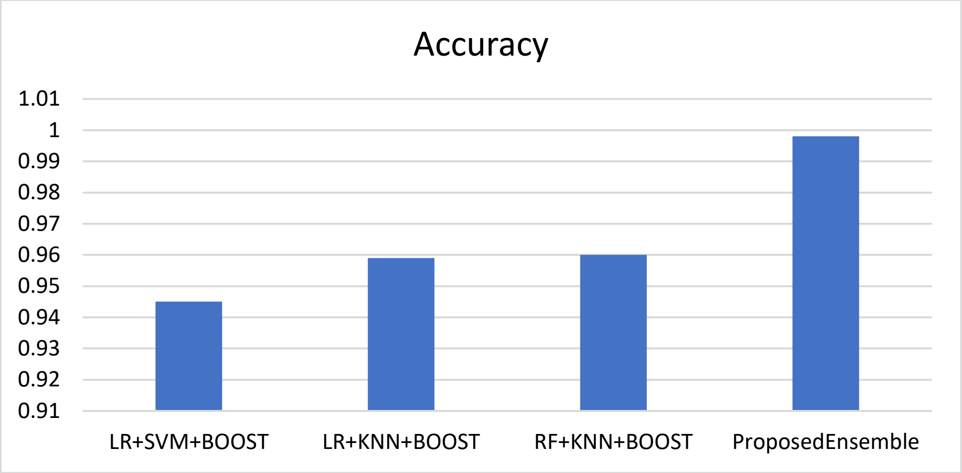
,

The above result represent the test samples classification accuracy of the proposed model on the selected features subset using ensemble learning framework. From the results it is noted that the proposed ranked based classification has better efficiency such as recall , true positive rate , precision etc on SSDS data.

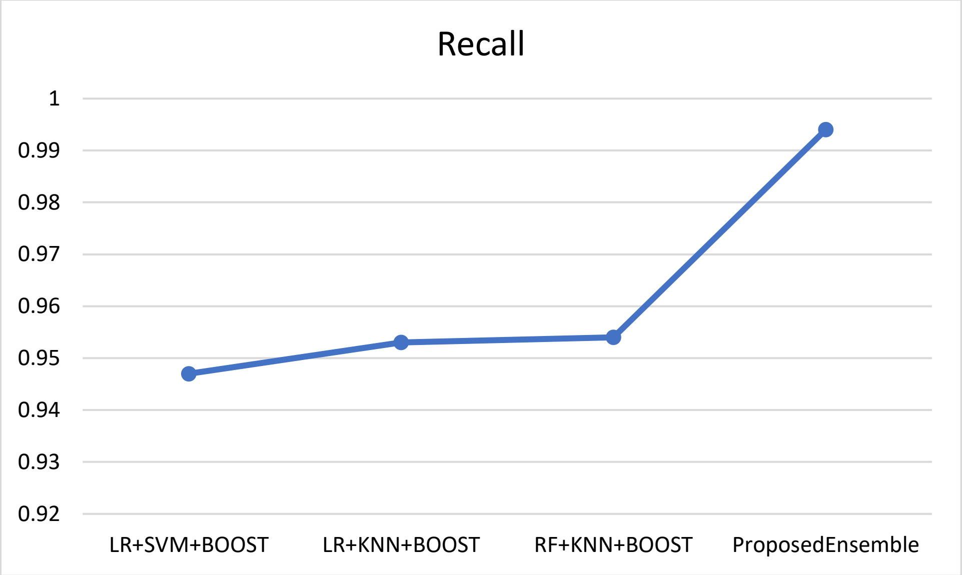


```
=== Stratified cross-validation ===

Correctly Classified Instances         550               99.6377 %
Incorrectly Classified Instances         2                0.3623 %
Kappa statistic                          0.8732
Mean absolute error                      0.007
Root mean squared error                  0.062
Relative absolute error                 22.9143 %
Root relative squared error             51.8811 %
Total Number of Instances              552

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.998    0.125    0.998      0.998    0.998      0.873    0.880     0.995     no
                 0.875    0.002    0.875      0.875    0.875      0.873    0.880     0.721     yes
Weighted Avg.    0.996    0.123    0.996      0.996    0.996      0.873    0.880     0.991
```

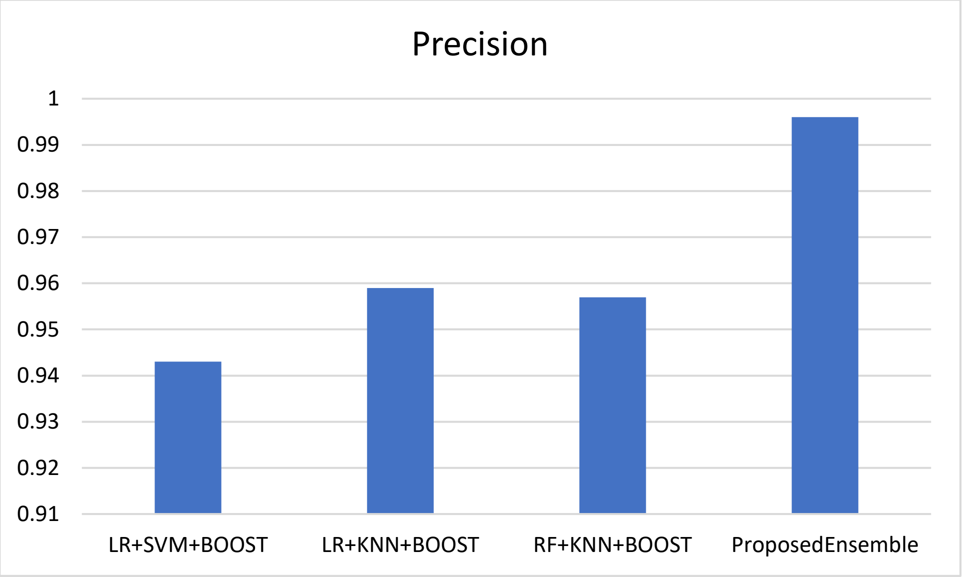
The above result represent the test samples classification accuracy of the proposed model on the selected features subset using ensemble learning framework. From the results it is noted that the proposed ranked based classification has better efficiency such as recall , true positive rate , precision etc on SSDS data.



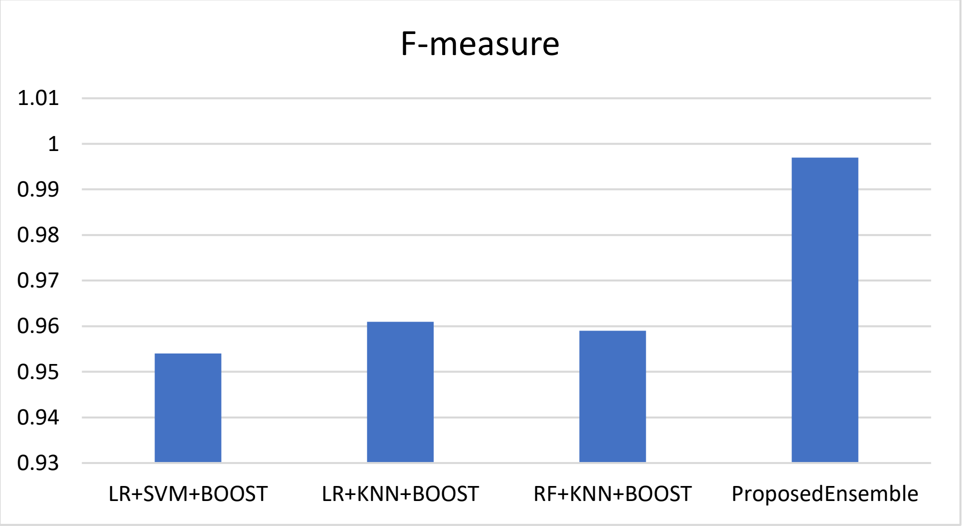
The above result represent the test classification accuracy of the proposed model on the selected features subset using ensemble learning framework. From the results it is noted that the proposed ranked based classification has better accuracy than conventional approaches on SSDS data.



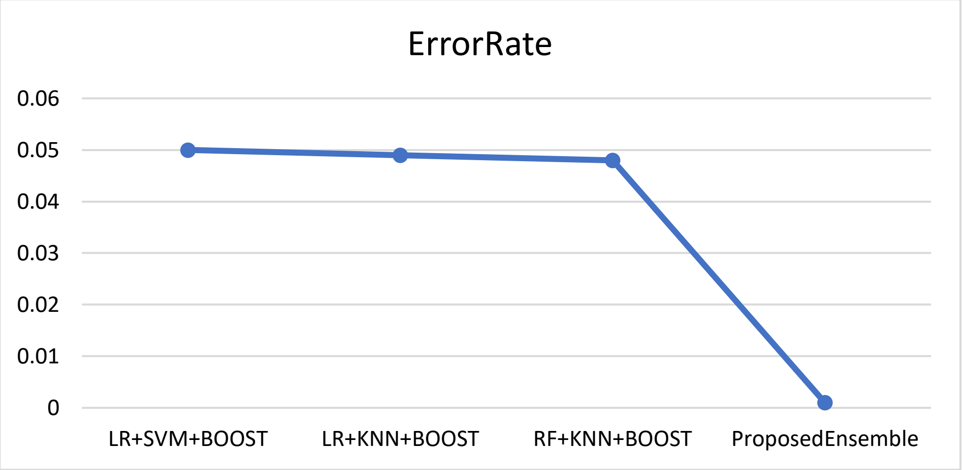
The above result represent the test classification recall of the proposed model on the selected features subset using ensemble learning framework. From the results it is noted that the proposed ranked based classification has better recall than conventional approaches on SSDS data..



The above result represent the test classification precision of the proposed model on the selected features subset using ensemble learning framework. From the results it is noted that the proposed ranked based classification has better precision than conventional approaches on SSDS data..



The above result represent the test classification F-measure of the proposed model on the selected features subset using ensemble learning framework. From the results it is noted that the proposed ranked based classification has better F-measure than conventional approaches on SSDS data..



The above result represent the test classification error rate of the proposed model on the selected features subset using ensemble learning framework. From the results it is noted that the proposed ranked based classification has better error rate than conventional approaches on SSDS data..

5. Conclusion

In conclusion, there are various techniques and methods available for privacy preserving machine learning (PPML), such as differential privacy, homomorphic encryption, secure multiparty computation, and data perturbation. Each of these techniques has their own benefits and challenges, and the choice of technique will depend on the specific requirements of the task at hand. One of the key challenges in PPML is the trade-off between privacy and accuracy. As more privacy preserving techniques are used, the accuracy of the machine learning algorithms may be compromised. Additionally, the scalability of the proposed solutions can be a challenge when dealing with large amounts of data. Another important challenge in PPML is the issue of data privacy in distributed systems. As data is distributed across multiple parties, it becomes increasingly difficult to ensure that it is protected from unauthorized access. Researchers have proposed various solutions to this problem, such as the use of secure multiparty computation to perform computations on encrypted data. When evaluating the performance of models trained on protected data, it is important to consider various metrics that can provide insight into the effectiveness of the privacy preserving technique. These metrics can include accuracy, precision, recall, and F-measure, which are commonly used in the context of classification tasks. Furthermore, comparing the models trained on protected data with those trained on non-protected data is a good way to demonstrate the effectiveness of the privacy preserving technique in maintaining the privacy of the individuals while still allowing for accurate analysis. In recent years, there has been a growing interest in the application of PPML in healthcare, as it has the potential to revolutionize the way medical data is collected, stored, and analyzed.

References

[1] N. Wang et al., “A blockchain based privacy-preserving federated learning scheme for Internet of Vehicles,” Digital Communications and Networks, May 2022, doi: 10.1016/j.dcan.2022.05.020.

[2] D. G. Nair, J. J. Nair, K. Jaideep Reddy, and C. V. Aswartha Narayana, “A privacy preserving diagnostic collaboration framework for facial paralysis using federated learning,” Engineering Applications of Artificial Intelligence, vol. 116, p. 105476, Nov. 2022, doi: 10.1016/j.engappai.2022.105476.

[3] W. Wang, X. Li, X. Qiu, X. Zhang, V. Brusic, and J. Zhao, “A privacy preserving framework for federated learning in smart healthcare systems,” Information Processing & Management, vol. 60, no. 1, p. 103167, Jan. 2023, doi: 10.1016/j.ipm.2022.103167.

[4] J. Zhao et al., “CORK: A privacy-preserving and lossless federated learning scheme for deep neural network,” Information Sciences, vol. 603, pp. 190–209, Jul. 2022, doi: 10.1016/j.ins.2022.04.052.

[5] C. Dhasarathan et al., “COVID-19 health data analysis and personal data preserving: A homomorphic privacy enforcement approach,” Computer Communications, vol. 199, pp. 87–97, Feb. 2023, doi: 10.1016/j.comcom.2022.12.004.

[6] S. Srijayanthi and T. Sethukarasi, “Design of privacy preserving model based on clustering involved anonymization along with feature selection,” Computers & Security, vol. 126, p. 103027, Mar. 2023, doi: 10.1016/j.cose.2022.103027.

[7] 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.

[8] Y. Wang, J. Ma, N. Gao, Q. Wen, L. Sun, and H. Guo, “Federated fuzzy k-means for privacy-preserving behavior analysis in smart grids,” Applied Energy, vol. 331, p. 120396, Feb. 2023, doi: 10.1016/j.apenergy.2022.120396.

[9] N. Rodríguez-Barroso et al., “Federated Learning and Differential Privacy: Software tools analysis, the Sherpa.ai FL framework and methodological guidelines for preserving data privacy,” Information Fusion, vol. 64, pp. 270–292, Dec. 2020, doi: 10.1016/j.inffus.2020.07.009.

[10] M. Khan, F. G. Glavin, and M. Nickles, “Federated Learning as a Privacy Solution - An Overview,” Procedia Computer Science, vol. 217, pp. 316–325, Jan. 2023, doi: 10.1016/j.procs.2022.12.227. [11] S. K. Singh, L. T. Yang, and J. H. Park, “FusionFedBlock: Fusion of blockchain and federated learning to preserve privacy in industry 5.0,” Information Fusion, vol. 90, pp. 233–240, Feb. 2023, doi: 10.1016/j.inffus.2022.09.027.

[12] J. Zhang, Y. Huang, Q. Huang, Y. Li, and X. Ye, “Hasse sensitivity level: A sensitivity-aware trajectory privacy-enhanced framework with Reinforcement Learning,” Future Generation Computer Systems, vol. 142, pp. 301–313, May 2023, doi: 10.1016/j.future.2023.01.008.

[13] 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.

[14] 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.

[15] 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.

[16] X. Zhu, J. Wang, W. Chen, and K. Sato, “Model compression and privacy preserving framework for federated learning,” Future Generation Computer Systems, vol. 140, pp. 376–389, Mar. 2023, doi: 10.1016/j.future.2022.10.026.

[17] 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.

[18] A. K. Nair, J. Sahoo, and E. D. Raj, “Privacy preserving Federated Learning framework for IoMT based big data analysis using edge computing,” Computer Standards & Interfaces, vol. 86, p. 103720, Aug. 2023, doi: 10.1016/j.csi.2023.103720.

[19] R. Venugopal, N. Shafqat, I. Venugopal, B. M. J. Tillbury, H. D. Stafford, and A. Bourazeri, “Privacy preserving Generative Adversarial Networks to model Electronic Health Records,” Neural Networks, vol. 153, pp. 339–348, Sep. 2022, doi: 10.1016/j.neunet.2022.06.022.

[20] Y. Wan, Y. Qu, L. Gao, and Y. Xiang, “Privacy-preserving blockchain-enabled federated learning for B5G-Driven edge computing,” Computer Networks, vol. 204, p. 108671, Feb. 2022, doi: 10.1016/j.comnet.2021.108671.