Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.022276
Article

Tech Science Press

A Secure Encrypted Classified Electronic Healthcare Data for Public Cloud Environment

Kirupa Shankar Komathi Maathavan[1,*] and Santhi Venkatraman[2]

[1]Department of Computer Science and Engineering, Government College of Engineering, Dharmapuri, 636704, Tamilnadu, India
[2]Department of Computer Science and Engineering, PSG College of Technology, Coimbatore, 641004, Tamilnadu, India
*Corresponding Author: Kirupa Shankar Komathi Maathavan. Email:kirupaa1991@gmail.com

Received: 02 August 2021; Accepted: 03 September 2021

Abstract: The major operation of the blood bank supply chain is to estimate the demand, perform inventory management and distribute adequate blood for the needs. The proliferation of big data in the blood bank supply chain and data management needs an intelligent, automated system to classify the essential data so that the requests can be handled easily with less human intervention. Big data in the blood bank domain refers to the collection, organization, and analysis of large volumes of data to obtain useful information. For this purpose, in this research work we have employed machine learning techniques to find a better classification model for blood bank data. At the same time, it is vital to manage data storage requirements. The Cloud offers wide benefits for data storage and the simple, efficient technology is adapted in various domains. However, the data to be stored in the cloud should be secured in order to avoid data breaches. For this, a data encryption module has been incorporated into this research work. The combined model provides secure encrypted classified data to be stored in the cloud, which reduces human intervention and analysis time. Machine learning models such as Support Vector Machine (SVM), Multinomial Naive Bayes (MNB), Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), K-Nearest Neighbor (KNN) are used for classification. For data security, the Advanced Encryption Standard with Galois/Counter Mode (AES–GCM) encryption model is employed, which provides maximum security with minimum encryption time. Experimental results demonstrate the performance of machine learning and encryption techniques by processing blood bank data.

Keywords: Electronic health records (EHR); big data; classification; machine learning; data security; encryption; cloud

1 Introduction

World Health organization (WHO) reports that on an average around 118.5 million blood donations happened globally in 2018. The report covers that 72% or 123 out of 171 countries had a national blood policy. From 2013 to 2018 the rate of blood donation has increased into 7.8 million which is reported by 156 countries. Handling these large volumes of data essentially needs an efficient processing system.

The recent advancement and technology development in information and communication has step into all the sectors in the world. Particularly, information technology in health sector wipes out the traditional methodologies for data management and provides innovative solutions to handle the patient data efficiently [1].

The blood data management systems need human interventions in order to analyze the data for future use. Most blood banks are still running manual system which includes paper-based information collection about donors, blood bags inventories and transfusion services. The manual analysis requires more time and chances of errors is large due to large volume of data. These time consuming and manual data management are eradicated in the digital era. Technology development reduces the human efforts and improves the diagnosis precision in the healthcare sector due to digital technologies.

Though the healthcare records are digitized still it requires human intervention to analyze the data. Medical data analysis needs high precision and accuracy so that further issues can be eliminated [2]. The blood data management analysis can be categorized into two modules. The first one is pure technical which essentially manages the data related to blood samples after processing the sample. The second one majorly deals the user data such as personal information, sample collection location, data. Analysis of these user data can be helpful to utilize the same person in future case if there is a required of blood. For this purpose, data analyzers are introduced in healthcare domain which classifies the sensitive data into different classes. Machine learning techniques are one among them which is widely used for various classification and clustering approaches in image processing applications [3]. Whereas in healthcare data analysis, machine learning models are employed in recent years. The sensitive user data can be identified and classified using machine learning techniques reduces the human intervention and errors in data management.

While machine learning gains more attention in healthcare data analysis, cloud computing transfers the medical data analysis into next level as virtual storage and ease access of healthcare data. The rapid growth of huge amount of data needs an efficient platform to handle and process the data [4]. Cloud offers numerous benefits and the virtual resources can store larger amount of data. Due to these benefits, Electronic Health Records (EHR) are moved into cloud platform. However, the same digital platform introduces numerous security and privacy challenges. Specifically, in healthcare data the user privacy is a major concern and preserving the user privacy from security attacks is a crucial task.

Cloud services are categorized into public cloud, private cloud, and hybrid cloud. Most of the healthcare data management systems employs public cloud which cannot fully be trusted by users [5]. The data outsourced in cloud are sensitive so privacy and security becomes major concern while deploying cloud services for EHR. Cloud offers several security measures to ensure the privacy and security of user data. However, from user side there is no such security measure so while transferring data to cloud it can be accessed. To prevent this, the data is encrypted in the user end and then transfer it to cloud is the only solution. Various encryption algorithms are evolved for data encryption however, it is essential an encryption algorithm should provide maximum security with minimum computation and communication cost.

The research work objective is framed to analyze electronic healthcare registers through machine learning techniques. Followed by classification, an efficient encryption for healthcare data is obtained to ensure the user data security and privacy. Finally, the encrypted data is moved into public cloud environment. Data collected from blood banks are analyzed through machine learning algorithms and based on the results the better performance machine learning model results are encrypted using Advanced Encryption Standard (AES) encryption with Galois/Counter Mode (GCM) for enhanced security.

2 Related Works

The rate of healthcare data increases rapidly and handling the data manually is a tedious process. It might consume more time also increases the probability of erroneous in the final results. To avoid this issue, healthcare records are converted digitally as EHR and stored in a data repository [6]. Feature selection is an important process in healthcare data analysis. Based on the selected features, classification is performed so that individual risk and preventive measures can be provided [7,8]. Similar to feature selection, feature reduction is also an important process in healthcare data analysis [9]. Processing huge volume of data will increase the computational complexity of the system. Feature reduction in healthcare data analysis reduces the error rate and increases the performance of the system. Numerous ML applications are applied in the healthcare data analysis [10–15] for classifying the medical data.

Based on the classification results a suitable decision can be obtained which reduces the extra burden of physicians. However, the healthcare records have various sensitive and user privacy information. It is essential to identify the sensitive data and preserve the user privacy is essential. ML techniques can be utilized to categorize the data into sensitive and non-sensitive data, so that user data security methodologies can be included in the data management process. A multi-source ordered preserving encryption for cloud-based eHealth system reported in [16] identifies the threats like frequency analysis, identical data inference and privacy leakage. An enhanced model of Merkle Hash Tree for multicopy storage of electronic medical records is reported in [17] that prevents data loss, unauthorized access to the sensitive user data. Lower communication and computation cost are considered as the merits of the research work. Sensitive and Energetic Access Control (SE-AC) mechanism presented in [18] ensures the data confidentiality of electronic health records and prevents authorized access. Secure query of personal health care data in cloud computing approach reported in [19]. A secure data storage algorithm reported in [20] employs Huffman compression and RC4 to reduce the ciphertext data amount and increases the data confidentiality. Attribute-based encryption (ABE) to secure personal health records in cloud computing is reported in [21] provides security to the health care records and maximizes the patient privacy. The security problems in Ciphertext-Policy Attribute-Based Encryption (CP-ABE) for electronic health record is reported in [22–24].

From the analysis, it is observed that encryption techniques are applied before transferring the data to cloud environment in order to secure the data and maintain the user privacy. ABE and CP-ABE is widely used in most of the research work for data encryption. However, CP-ABE has limitations in its access policy storage. Since it is encrypted in the ciphertext the possibility of policy leak may lead to security issues. Few research works address the methodologies to hide the access policies but these processes require an outsource policy manager which increases the setup cost. In the case of ABE, the performance is better however it can be further improved in terms of computation time. The majority of healthcare data analysis based on machine learning classifies the data for further diagnosis. on the other hand, the classification of data into sensitive and non-sensitive categories is not addressed earlier. Based on these findings, this research work proposes a data classification methodology for identifying sensitive information in blood bank data and encrypting it for further protection. As a result, when storing data in a public cloud environment, user data privacy and security are improved. This research work is novel as it provides data sensitivity analysis specifically for blood bank data, which has not been done before. Encrypting sensitive blood bank data is also performed in this research work to add the distinctiveness.

3 Proposed Work

The proposed health care data classification using machine learning techniques and encryption before transferred into cloud is discussed in this section. Fig. 1 depicts a simple illustration of proposed model process flow. The process starts from collecting manual data from the blood banks and convert into digital data. Before classification, preprocessing is performed. In the preprocessing, feature transformation, feature construction, data rows aggregation and tables combination are performed. In the feature transformation process, the discrete categorical features are replaced with numerical values with intervals. In feature construction, in order to extend the features of the original dataset some new set of features are included. This process reduces the prediction error while using the actual dataset.

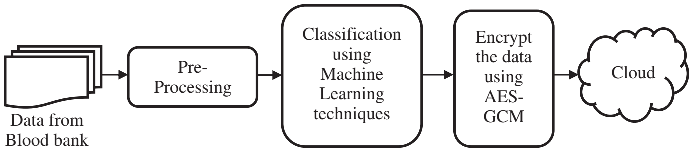

Data from Blood bank
Pre-Processing
Classification using Machine Learning techniques
Encrypt the data using AES-GCM
Cloud

Figure 1: Process overview of proposed model

Data rows aggregation in the preprocessing aggregates all the data in a single row. So that the dataset becomes easy to access and analyze. Finally, tables combination in the preprocessing step all the relevant data features that are collected for proposed work database are combined as a single table. This process reduces the computational complexity of the classifier models on training process. Machine learning models are employed as the classifier for the proposed approach. The classified results obtained from the machine learning models are further encrypted and stored in the public cloud using AES-GCM encryption technique for future access.

Machine learning techniques has an ability to automatically obtain deep insights of data and identify hidden patterns. The predictive model processes the data without the necessity of explicit programming instructions and human interventions. Machine learning is divided into supervised, unsupervised, and reinforcement learning and it is extensively utilized in many fields, including healthcare data analysis. In order to make the health care system into valuable and efficient with minimum human effort, machine learning techniques such as MNB, SVM, RF, DT, KNN, GB are widely used. The above-mentioned machine learning techniques are employed in the proposed work and the best model is selected based on the classification results.

3.1 Support Vector Machine

SVM is simple and efficient algorithm which is extensively used for classification applications. The classes in the dataset are categorized depends on the multidimensional hyperplane by maximizing the margin between the two clusters. The nonlinear functions called kernels supports the system model to achieve maximum discriminative power by transforming the input space into multidimensional space. For an n-dimensional space, the two data classes are discriminated based on the hyperplane in the SVM model.

3.2 Multinomial Naive Bayes

MNB is a popular supervised machine learning approach that provides better performance for text data categorization. The algorithm works based on Bayes theorem and predicts the tags based on the probability function. The highest probability tag will be considered as output of multinomial Naïve Bayes model. The output classified results in multinomial Naïve Bayes approach has unique features and it doesn’t relate any other features in it.

3.3 Decision Tree

The DT algorithm is a mapping model in which the classes are defined by linking the attributes into multiple sub-trees and leaves. The final decision nodes with classes are considered as decisions. Based on the attributes for an instance, test node computes the outcomes. Subtrees in the DT algorithm are used to represent each possible outcome.

3.4 Random Forest

A RF is an ensemble of independent DT in which single tree is replaced with ensemble of decorrelated trees to obtain good generalization. In the learning phase, to obtain independence between constructed trees, a randomness can be included in the RF approach. RF can be used to obtain better solution for task specific objective functions with various classes of posterior functions.

3.5 Gradient Boosting

The GB is a prediction model which is generally used for classification, regression and other tasks. Typically, it is similar to DT. However, the function gives an approximate data distribution based on the vectors of predictors. Mathematically the function is expressed as

$$y_i = f_1(x_i) + e_i \tag{1}$$

where x_i represents the predictor vector and y_i represents the outcome variable. e_i represents the error.

3.6 K-Nearest Neighbor

The KNN is a generalization algorithm which performs analysis based on the nearest neighbor rules. Compared to nearest neighbor, KNN expands the nearest neighbor to k values in the decision-making process. It eliminates the learning process relative to other classification algorithms with training phases. The decision process of KNN is simple and efficient.

Once the data classification is completed using machine learning techniques, the best classification results are stored in the cloud for further process. However, the data to be stored in the cloud must be secured so data encryption is incorporated in the proposed work to secure the privacy information of donor and medical data. For this Advanced Encryption Standard (AES) algorithm is included with Galois/Counter Mode of operations as (AES-GCM) model. The processes in AES encryption are discussed in the following section.

3.7 AES Encryption

AES encryption algorithm is a symmetric block cipher encryption algorithm which is attained from the substitution permutation network. It uses the same key for both encryption and decryption. The block of cipher key size in AES is generally 128 bits and depends on application it can be extended to 192 bits and 256 bits. Depends on the size of the block, key length, the number of rounds will be given as 14 for 256 bits, 12 for 192 bits and 10 for 128 bits. Tab. 1 depicts the key size and its respective number of rounds in detail.

Table 1: Number of rounds and key size

AES key (bits)	Size of the block	Key length	Number of rounds
128	4	4	10
192	4	6	12
256	4	8	14

The 128 bits are generally arranged into a 4 × 4 matrix. Fig. 2 gives an illustration of AES encryption algorithm. The process in AES encryption is divided into four such as byte substitution, mix columns, shift rows, and finally add round key. In the byte substitution each byte is replaced with a new byte and the plain text values are replaced with the substitution box values as shown in Fig. 3.

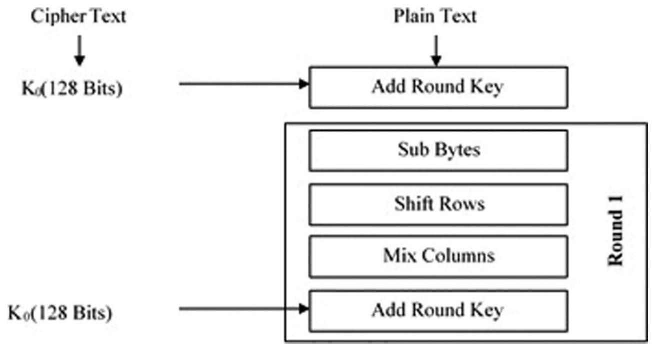


Figure 2: AES process flow

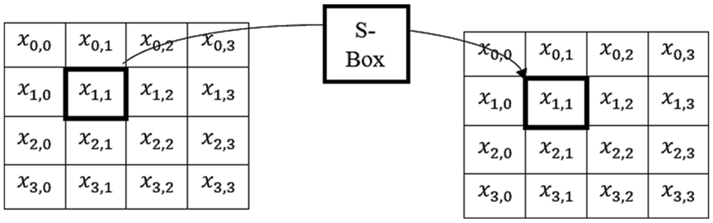

$x_{0,0}$ $x_{0,1}$ $x_{0,2}$ $x_{0,3}$
$x_{1,0}$ $x_{1,1}$ $x_{1,2}$ $x_{1,3}$
$x_{2,0}$ $x_{2,1}$ $x_{2,2}$ $x_{2,3}$
$x_{3,0}$ $x_{3,1}$ $x_{3,2}$ $x_{3,3}$
S-Box
$x_{0,0}$ $x_{0,1}$ $x_{0,2}$ $x_{0,3}$
$x_{1,0}$ $x_{1,1}$ $x_{1,2}$ $x_{1,3}$
$x_{2,0}$ $x_{2,1}$ $x_{2,2}$ $x_{2,3}$
$x_{3,0}$ $x_{3,1}$ $x_{3,2}$ $x_{3,3}$

Figure 3: Byte substitution in AES encryption

The shift operation depicted in Fig. 4 is a permutation process in which the data is shifted to left side in an incremental order for each element. Suppose the shift operation is performed for the first row, then the blocks will shift into left side for one position, if the shift operation is performed for second row, then two blocks will be shifted into left side. Similarly, the process continues for all the rows. Generally, nth row elements will shift into n times in the shift operation.

Figure 4: Shift operation

Followed by shift operation, the columns are mixed in the next stage. The first column in the first matrix is mixed with the first column in the second matrix and this process repeats for all the columns. Fig. 5 demonstrates the mix column operation in detail.

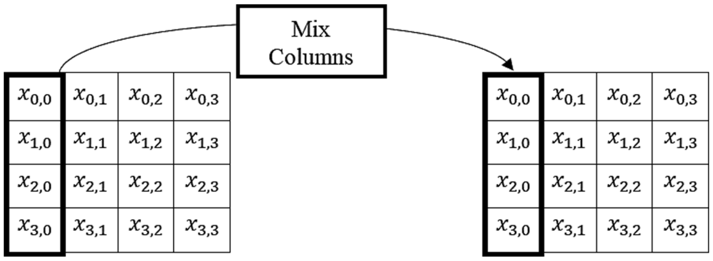
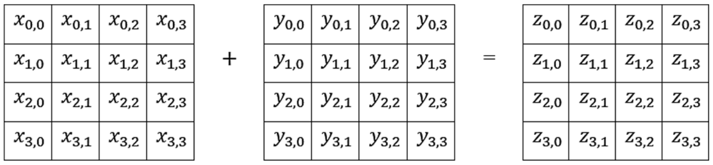

Mix Columns
$x_{0,0}$ $x_{0,1}$ $x_{0,2}$ $x_{0,3}$
$x_{1,0}$ $x_{1,1}$ $x_{1,2}$ $x_{1,3}$
$x_{2,0}$ $x_{2,1}$ $x_{2,2}$ $x_{2,3}$
$x_{3,0}$ $x_{3,1}$ $x_{3,2}$ $x_{3,3}$
$x_{0,0}$ $x_{0,1}$ $x_{0,2}$ $x_{0,3}$
$x_{1,0}$ $x_{1,1}$ $x_{1,2}$ $x_{1,3}$
$x_{2,0}$ $x_{2,1}$ $x_{2,2}$ $x_{2,3}$
$x_{3,0}$ $x_{3,1}$ $x_{3,2}$ $x_{3,3}$

Figure 5: Mix column operation

An add around key is added with the matrix values obtained after mix column operation. Logical XOR operation is performed in the addition process which XORs the 16-byte expanded key and plain text. The final matrix after XOR operation is the encrypted text. Fig. 6 depicts the add around key operation in detail.



$$\begin{array}{|c|c|c|c|}\hline x_{0,0} & x_{0,1} & x_{0,2} & x_{0,3} \\ \hline x_{1,0} & x_{1,1} & x_{1,2} & x_{1,3} \\ \hline x_{2,0} & x_{2,1} & x_{2,2} & x_{2,3} \\ \hline x_{3,0} & x_{3,1} & x_{3,2} & x_{3,3} \\ \hline \end{array} + \begin{array}{|c|c|c|c|}\hline y_{0,0} & y_{0,1} & y_{0,2} & y_{0,3} \\ \hline y_{1,0} & y_{1,1} & y_{1,2} & y_{1,3} \\ \hline y_{2,0} & y_{2,1} & y_{2,2} & y_{2,3} \\ \hline y_{3,0} & y_{3,1} & y_{3,2} & y_{3,3} \\ \hline \end{array} = \begin{array}{|c|c|c|c|}\hline z_{0,0} & z_{0,1} & z_{0,2} & z_{0,3} \\ \hline z_{1,0} & z_{1,1} & z_{1,2} & z_{1,3} \\ \hline z_{2,0} & z_{2,1} & z_{2,2} & z_{2,3} \\ \hline z_{3,0} & z_{3,1} & z_{3,2} & z_{3,3} \\ \hline \end{array}$$

Figure 6: Add around key operation

Galois/Counter mode (GCM) is a block cipher mode of operation. It is introduced to obtain authenticated encryption model for the binary Galois field which utilizes universal hashing. High speed, low latency and low cost are the major benefits of GCM and hardware implementation is possible in it. Similar to hardware performances the software implementation will also provide better performances. The table-driven field operations support the software implementations. The theoretical foundation and security provide reasonable assumption about the block cipher. Fig. 7 depicts the working model of AES-GCM encryption. During encryption of decryption counter model is selected to process the incoming data. In the authentication process a tag up to 128 bits is produced by hashing with Galois fields accumulator data as well as the result of counter mode encryption and decryption. In the authentication process, if the user tag matches with original then decryption is allowed otherwise the authentication gets failed and no other information can be decrypted from the database. The GCM can be simply expressed as a function of secret key, initialization vector, plain text and authentication data as follows

$$(T, c) = GCM(k_s,\ v_{ini},\ p_t,\ Ad_{data}) \tag{2}$$

where k_s represents the secret key used to encrypt the data, v_{ini} represents the initialization vector, p_t represents the plain text and Ad_{data} represents the additional authentication data. The overall function is defined with the parameters T and c which represent the authenticated tag used in the decryption process that verifies the data authenticity and encrypted information respectively. The encryption and authentication formulations for GCM is given as

$$y_{i+1} = inc(y_i) \quad for\ i = 1, 2, \ldots., n \tag{3}$$

$$c_i = p_i \oplus e(k_s, y_i) \quad for\ i = 1, 2, \ldots., n \tag{4}$$

$$T = MSB_t(G_{hash}(H, A, C) \oplus e(k_s, y_0)) \tag{5}$$

where $e(k_s, y_i)$ represents the AES counter mode encryption function, the hashing function G_{hash} used for authentication and MSB_t represents most significant bit which is considered for hashing function. The hashing algorithm is simply expressed as

$$x_{i+1} = (input + x_i) * H \tag{6}$$

The hashing function is a simple 128-bit XOR operation and the Galois multiplication which is used as feedback limits the functions of G_{hash}. The Galois multiplication process is different from conventional base 10 addition and multiplication. The carry less operation in Galois addition or multiplication does not increase the bit size. Due to this carry less operation, addition and subtraction of two polynomials will produce same results and because of this feature, Galois process is more suitable for software and hardware operations.

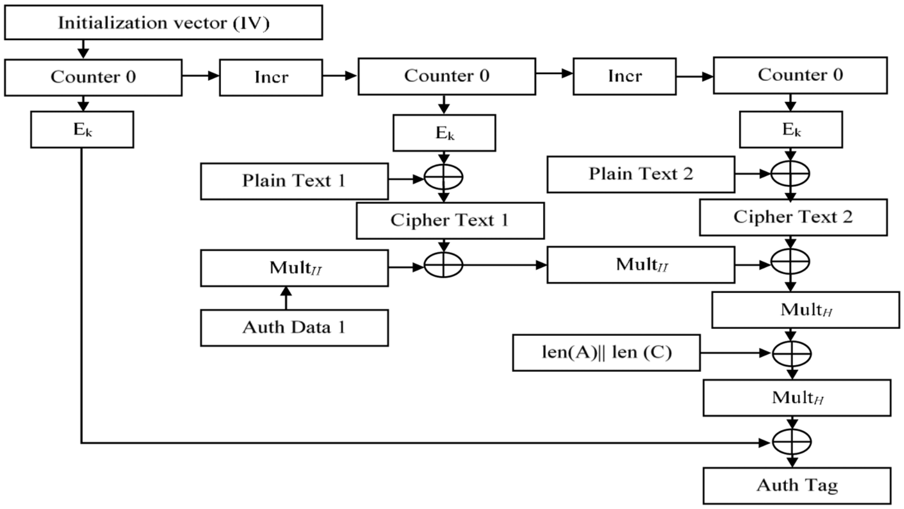


Figure 7: AES-GCM working model

The system needs to be initialized whenever a new key and initialization vector is received. For that, it is essential to prepare the hash key which can be obtained from the AES encryption output. Once the initialization is done, the system is ready to accept input for encryption or additional data. Generally, the additional data is preceded by encryption data. summarized pseudocode for encryption and decryption using AES-GCM algorithm is given as follows.

Algorithm 1: Pseudocode for AES-GCM

```
Input: Key stream, tag, seg ID, timestamp
Output: encrypted and decrypted data
Begin encryption
Obtain cipherkey from AES encryption performing byte substation, mix columns, shift rows, and add round key operations
Obtain timestamp and tag using GCM
Define the function {key stream, tag} = AES-GCM (segment ID, Data address, time stamp)
Obtain cipher text = plain text $\oplus$ keystream
Transfer new ciphertext into cloud storage
Update timestamp using memory address
Tag authentication storage using tag memory address
End encryption
Begin decryption
Initialize time stamp from memory address
Initialize tag authentication from tag memory address
Decrypt using {key stream, tag} = AES-GCM (segment ID, Data address, time stamp)
```

(Continued)

Algorithm 1 (continued)

```
Load ciphertext from external cloud memory
Obtain plain text = cipher text ⊕ keystream
Check authentication if (tag = tag)
Obtain plain text from memory
Else restrict the access
End
```

4 Results and Discussion

Performance of the proposed classification and encryption model is experimentally analyzed in this section. The results demonstrate the performance of better classification and security model for blood bank data management. Experimentations are performed in Python and for machine learning algorithms, numpy, pandas, sklearn libraries are included. For encryption PyCryptodome library has been used. The secure encrypted data are stored in Azure Cosmos database. The system configuration used for experimentation is Intel i3 7th gen processor 3.9 GHz with 8 GB RAM.

The dataset used for the experimentation is a real time data which is obtained from the KSM Blood Bank, Salem, Tamilnadu, India and Kongu Blood Bank, Erode, Tamilnadu, India. Details such as data of donation, donor name and address, unit number, date of collection and expiry, blood group, quantity, results after analysis like antibodies presence, HIV status, HCV, HBsAg, VDRL, MP and date of component prepared along with other extra details are collected for a duration of 10 months. The dataset includes 1,00,000 samples which is obtained from the blood banks. The data is split into 80:20 ratio for training, testing respectively and for validation fivefold cross is used in the experimentation. The initial preprocessing steps is performed as common step and for classification different machine learning algorithms such as SVM, MNB, RF, KNN, DT, GB are used. From these results, best classification model is selected for next step encryption process.

Tab. 2 depicts the features used for experimentation and its description for better understanding. The attributes are selected from the features. initially depends on the data type the attribute is labeled into 1 for integer, 2 for floating number, 3 for string and 4 for date and time. The null percentage attribute defines percentage of non-missing values in each column. It is the ratio of actual number of values to the total number of values. Unique percentage defines the number of unique values obtained in the feature. As we are using blood bank data, the user personal identification data will be different for each other and these unique attributes are considered for sensitive data analysis. Finally, for pattern-based sensitivity the attributes are labelled into 0 for "No" and 1 for "yes". Regular expression is used to analyze the sensitivity.

Table 2: Features used in the proposed classification model

S. No	Feature	Type	Description
1	Data type	Numerical	1 for integer, 2 for floating number, 3 for string, 4 for date time
2	Null percentage	Integer	Percentage of NULL values in a column
3	Unique percentage	Numerical	Percentage of unique values in a column
4	Pattern based sensitivity	Binary	Pattern based Sensitivity analysis using regular expression, 0 for no and 1 for yes.

The selected attributes are provided as input to the machine learning models. The analysis is performed as two experimentations. In this first experimentation, two features are considered and its performance is measured for all the machine learning approaches. In the second experimentation, four features are considered for analysis and its performances are measured. The reason for these two experimentations is to demonstrate the importance of data analyzer and its performances. Since with minimal features, any classification model can provide better results, but those results will not be more accurate, so to highlight the importance of more feature-based classification, these two cases of experimentation are presented in this research work. The following parameters are calculated to measure the performance of machine learning models.

$$Accuracy = \frac{True\ Positive\ +\ True\ Negative}{Positive\ +\ Negative} \tag{7}$$

$$Precision = \frac{True\ Positive}{True\ Positive\ +\ False\ Positive} \tag{8}$$

$$Recall = \frac{True\ Positive}{True\ Positive\ +\ False\ Negative} \tag{9}$$

$$F_1\ Score = \frac{True\ Positive}{True\ Positive\ +\ \frac{1}{2}\left(False\ Positive +\ False\ Negative\right)} \tag{10}$$

Case 1: Classification Based on Two Features

In the first case, two features are considered for analysis. The collected data from the blood bank is converted into two features such as frequency and sensitivity. Based on that the classes are allocated for each donor. 20% of testing samples are used for analysis and its performance metrics are measured through various parameters as depicted in Tab. 3.

Table 3: Performance metrics for machine learning models for 2 feature analysis

Algorithm	Cross validation score (%)	Training accuracy (%)	Testing accuracy (%)	Correctness (%)
SVM	95	100	80	70
MNB	70	70.59	60	60
DT	100	100	100	65
RF	90	100	100	75
GB	100	100	100	60
KNN	65	94	80	65

The results for training and test accuracies are presented along with cross validation score and correctness score. It can be observed from the results the training and test score for DT, RF and GB are maximum whereas other models exhibit less performance due to less features. However, the cross-validation score for RF gets reduced in 90 similarly the correctness score for DT and GB are reduced into 65 and 60 respectively. From this it can be concluded that none of the machine learning models can able to classify the data accurately due to minimum features.

The performance metrics is further extended to measure other parameters like recall, precision, and F1 score for machine learning models and the comparative analysis is presented in Fig. 8. From the figure it can be observed that the performance of DT, RF, GB and KNN are maximum. Similarly, the performance of SVM and MNB attains less scores. Though the models attain maximum score it will not be considered as better results as the validation score and correctness factors are differs for each model. So, the analysis is further extended based on four features.

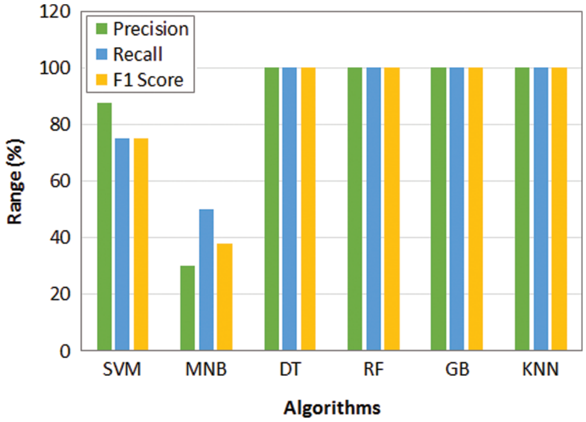


Figure 8: Performance comparison of machine learning models for 2 feature analysis

Case 2: Classification Based on Four Features

The classification results based on two features are not satisfactory, so to enhance the performance of machine learning algorithms, four features are considered in this case. The features such as data type, null percentage, unique percentage and Pattern based Sensitivity are considered and based on that sensitivity class is obtained. Tab. 4 depicts the performance metrics of machine learning models for cross validation score, training Score, testing score and correctness factor.

Table 4: Performance metrics for machine learning models for 4 feature analysis

Algorithm	Cross validation score (%)	Training accuracy (%)	Testing accuracy (%)	Correctness (%)
SVM	88	89	85	87
MNB	82	82	78	70
DT	90	100	95	87
RF	90	97	95	90
GB	86	92.37	90	73
KNN	76.47	80	100	84

It can be obtained from the results the training score is maximum for DT model and testing score is maximum for KNN. Where a maximum cross validation score is obtained by DT and RF. The maximum correctness score is obtained by RF model. From the results, it is clear that DT and RF methods perform better than other models. However, for detailed analysis, the performance metrics like precision, recall and f1 score are calculated for this case also and it is depicted in Fig. 9.

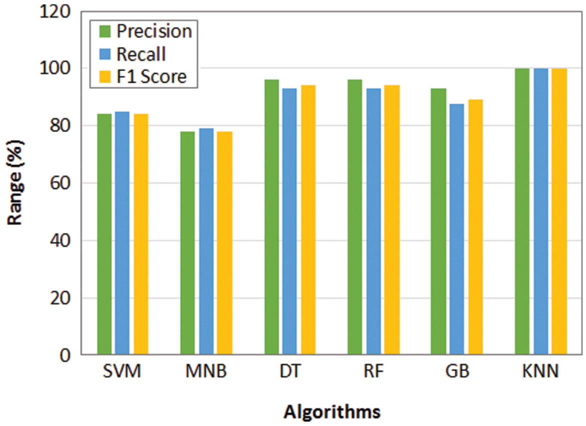


Figure 9: Performance comparison of machine learning models for 4 feature analysis

The performance metrics is further extended to measure other parameters like precision, recall and f1 score for machine learning models and the comparative analysis is presented in Fig. 10. It can be observed from the results, KNN model obtains maximum performance for precision, recall and F1 score. However, the training accuracy of the KNN is 80% which indicates the results are not more accurate as we expected. Whereas the training score for DT is 100% but it attains the testing score as 95% and this reflects in the other parameter's such as 96% of precision, 93% for recall and 94% for F1 score which is lesser than KNN. It can be observed from the analysis in both cases, the performance of SVM and MNB attains less scores. The hyperspace in the SVM categorizes the data into and due to large number of categories in the dataset, vector machine considers few important data as outliers so that the performance gets reduced compared to other models.

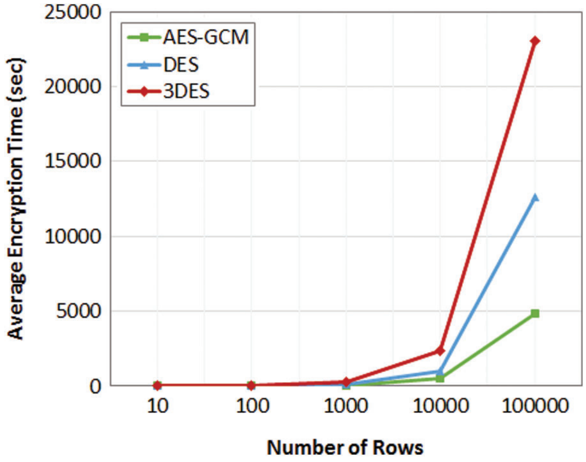


Figure 10: Average encryption time of encryption algorithms

Whereas MNB has poor learning ability reduces the overall performance in the sensitivity data classification. In case of GB, the performance reduces due to data overfitting and similar to SVM it also considers the features as outliers.

The performance of remaining models is considered to be better. In particular KNN attains maximum scores for precision, recall and F1 score. However, the training accuracy is 80% and testing accuracy is 100% clearly depicts that wrong classification. The testing accuracy is 95% for both DT and RF. But similar to KNN the training accuracy of RF is 97% so that it cannot be considered as accurate. So, from training and testing accuracy values, the performance of DT model is considered as much better than other models.

Once the data classification is performed the classified results are stored in a public cloud. For enhanced data security AES-GCM encryption is employed in the proposed work. To measure the performance of encryption model, encryption time for 1000 Bits and 10000 Bits are obtained.

To validate the superior performance of existing encryption methods such as 3DES, DES are compared with proposed AES-GCM and it is listed in Tab. 5. It can be observed from the results in Tab. 5, the proposed AES-GCM encryption model takes minimum encryption time of 0.19 s for 1000-bit data and 12.43 s for 10000-bit data. whereas the encryption time of other techniques is high compared to the proposed encryption model.

Table 5: Comparison of encryption techniques

Algorithms	Encryption time (Sec)		Secure	Data integrity
	1000 Bits	10000 Bits		
3DES	0.24	16.12	✗	✗
AES-GCM	0.19	12.43	✓	✓
DES	725.35	8363.28	✗	✗

The average encryption time comparison for all three encryption techniques is compared in Tab. 6. The time is measured for the different number of rows in the dataset from 10 to 100000 and its respective encryption time is measured. The performance of AES-GCM is better than DES algorithms as AES has different rounds of operations based on the key length, whereas DES involves fixed 16 round operations for all the keys it consumes more time than AES algorithm.

Table 6: Comparison of average encryption time encryption algorithms

Number of rows	AES (Sec)	DES (Sec)	3DES (Sec)
10	0.48	0.49	0.48
100	4.96	5.04	5.12
1000	48.35	63.21	88.68
10000	485.97	810.54	1345.34
100000	4839.57	12593.42	23053.64

Another reason for minimum computation time for AES is its ability to encrypt 128 bits of plain text, whereas DES can able to encrypt 64 bits of plain text. Due to the known vulnerabilities DES can be broken easily whereas AES is defined as per the global standards and it is difficult to break the security.

The observations are plotted in Fig. 10 and it can be observed from the figure the performance of proposed AES-GCM model is much better than DES and 3DES encryption techniques. Initially for smaller number of rows the performance of all the three algorithms are almost similar but when the number of rows is increased gradually the performance of DES and 3DES reduces as the encryption time increases. For maximum of 100000 rows of data the AES-GCM takes an average of 4839 s which is 2.6% lesser than DES model and 4.7% lesser than 3DES model. The overall computation time for the proposed secure encrypted electronic health data classification for public cloud is obtained and listed in Tab. 7. The computation time is measured when the system processing 10 rows of data, 10000 rows of data and 100000 rows of data.

Table 7: Comparison of overall computation time

S. No	Number of features	Total time (s) for 10 rows	Total time (s) for 10000 rows	Total time (s) for 100000 rows
1	2	1.69	1565.56	16442.42
2	4	1.53	1448.44	15824.36

The overall computation time is obtained for initial data analysis, data sensitivity classification, followed by encryption and storage of data in cloud environment. It can be observed from the data given in Tab. 7 the overall computation time based on 4 features is less compared to other models. Though the features are more the classification model classifies data quickly due to the final sensitive class. The encryption model also performs better than other models which further reduces the computation time. Due to these, the maximum performance is obtained for 4 feature-based analyses in case 2 compared to case 1.

5 Conclusion

This research work presents a secure encrypted classified electronic healthcare data for public cloud environment using machine learning and encryption techniques. The data management in blood bank supply chain and the difficulties in manual database maintenance are analyzed in this research work. The research model will be possible solution as automated healthcare data management system for blood banks. Different machine learning models are employed in the experimental analysis under two different cases. In each case, the number of features is changed and its performance metrics are observed. Among all the machine learning models, DT technique attains maximum performance. Further the data is stored in public cloud using AES-GCM encryption in order to secure the data. The performance metrics of encryption model is measured and the proposed encryption model completed the average encryption time in 4839.57 Seconds. This research work is fully focused on data classification and security. Though the performance of DT is better it can be further improved if the features are increased or an optimization model is included in the research work.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] N. A. Azeez and C. V. Vyver, “Security and privacy issues in e-health cloud-based system: A comprehensive content analysis,” *Egyptian Informatics Journal*, vol. 20, no. 2, pp. 97–108, 2018.

[2] R. Sahney and M. Sharma, “Electronic health records: A general overview,” *Current Medicine Research and Practice*, vol. 8, no. 2, pp. 67–70, 2018.

[3] M. R. Ahmed, Y. Zhang, Z. Feng, B. Lo, O. T. Inan *et al.,* "Neuroimaging and machine learning for dementia diagnosis: Recent advancements and future prospects," *IEEE Reviews in Biomedical Engineering*, vol. 12, pp. 19–33, 2018.

[4] L. Rajabion, A. A. Shaltooki and A. Badfar, "Healthcare big data processing mechanisms: The role of cloud Computing," *International Journal of Information Management*, vol. 49, no. 1, pp. 271–289, 2019.

[5] H. Wang, "Anonymous data sharing scheme in public cloud and its application in e-health record," *IEEE Access*, vol. 6, pp. 27818–27826, 2018.

[6] S. M. Shah and R. A. Khan, "Secondary use of electronic health record: Opportunities and challenges," *IEEE Access*, vol. 8, pp. 136947–136965, 2020.

[7] G. Tsang, S. M. Zhou and X. Xie, "Modeling large sparse data for feature selection: Hospital admission predictions of the dementia patients using primary care electronic health records," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 9, pp. 1–13, 2021.

[8] M. Moreb, T. A. Mohammed and O. Bayat, "A novel software engineering approach toward using machine learning for improving the efficiency of health systems," *IEEE Access*, vol. 8, pp. 23169–23178, 2020.

[9] S. J. Pasha and E. S. Mohamed, "Novel feature reduction (NFR) model with machine learning and data mining algorithms for effective disease risk prediction," *IEEE Access*, vol. 8, pp. 184087–184108, 2020.

[10] M. E. Hossain, A. Khan, M. A. Moni and S. Uddin, "Use of electronic health data for disease prediction: A comprehensive literature review," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 2, pp. 745–758, 2021.

[11] X. Pang, C. B. Forrest, F. L. Scherban and A. J. Masino, "Prediction of early childhood obesity with machine learning and electronic health record data," *International Journal of Medical Informatics*, vol. 150, no. 41, pp. 1–14, 2021.

[12] A. Garg and V. Mago, "Role of machine learning in medical research: A survey," *Computer Science Review*, vol. 40, pp. 1–16, 2021.

[13] F. Shamout, T. Zhu and D. A. Clifton, "Machine learning for clinical outcome prediction," *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 116–126, 2021.

[14] K. Y. Ngiam and I. W. Khor, "Big data and machine learning algorithms for health-care delivery," *The Lancet Oncology*, vol. 20, no. 6, pp. 262–273, 2019.

[15] J. Tohka and M. V. Gils, "Evaluation of machine learning algorithms for health and wellness applications: A tutorial," *Computers in Biology and Medicine*, vol. 132, pp. 1–14, 2021.

[16] J. Liang, Z. Qin and K. Li, "Privacy-preserving range query over multi-source electronic health records in public clouds," *Journal of Parallel and Distributed Computing*, vol. 135, no. 6, pp. 127–139, 2020.

[17] L. Zhou, A. Fu and Y. Sun, "Multicopy provable data possession scheme supporting data dynamics for cloud-based electronic medical record system," *Information Sciences*, vol. 545, no. 9, pp. 254–276, 2021.

[18] K. Ria, R. Hamza and H. Yan, "Sensitive and energetic IoT access control for managing cloud electronic health records," *IEEE Access*, vol. 7, pp. 86384–86393, 2019.

[19] X. Liu, Y. Xia and F. Yang, "Secure and efficient querying over personal health records in cloud computing," *Neurocomputing*, vol. 274, no. 2, pp. 99–105, 2018.

[20] J. Zhang, H. Liu and L. Ni, "A secure energy-saving communication and encrypted storage model based on RC4 for HER," *IEEE Access*, vol. 8, pp. 38995–39012, 2020.

[21] W. Li, B. M. Liu, D. Liu, R. P. Liu, P. Wang *et al.,* "Unified fine-grained access control for personal health records in cloud computing," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 3, pp. 1278–1289, 2019.

[22] Z. Ying, L. Wei, Q. Li, X. Liu and J. Cui, "A lightweight policy preserving EHR sharing scheme in the cloud," *IEEE Access*, vol. 6, pp. 53698–53708, 2018.

[23] Y. Zhang, D. Zheng and R. H. Deng, "Security and privacy in smart health: Efficient policy-hiding attribute-based access control," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2130–2145, 2018.

[24] S. Sharaf and N. F. Shilbayeh, "A secure G-cloud-based framework for government healthcare services," *IEEE Access*, vol. 7, pp. 37876–37882, 2019.