Privacy Preserving Model using Homomorphic Encryption

Asha Kiran Grandhi
Research Scholar
Savitribai Phule Pune University
Pune

Manimala Puri, PhD
Director
JSPM Group of Institutions
Pune

S. Srinivasa Suresh, PhD
Associate Professor
Business Analytics, KIAMS
Pune

ABSTRACT

Privacy preserving is utmost important in medical applications. Cryptography has numerous techniques to safe guard the privacy of the data. It is practice to use private key for encryption and public key for decryption in the area of cryptography. Conventionally, without decryption, data usability is difficult. However, the complications outweigh the private and public keys. This paper presents privacy preserving model based on Homomorphic encryption technique and model evaluation using classification technique. The homomorphic model highlights usability of the data without decryption. The objective of this paper is to show how the encrypted data is preserving underlying relations through classification tree. This paper presents two parts: Part-I describes the model building on medical data using PSO optimization and filer based co-efficient matrix (for encryption) to protect privacy of the data and part-II describes model evaluation using classification tree and clustering technique. The performance of the encryption is tested using predictive modelling technique (classification tree technique) and K-Means clustering technique, to assess whether the underlying relations are preserved in the encrypted data. The experimental results show that the underlying classification accuracy of encrypted data and source data (non-encrypted) is just varying by +/- 5%.

Keywords

Homomorphic Encryption, Predictive Modeling, Privacy Preserving Data Mining.

1. INTRODUCTION

Data privacy is the primary concern for the data owners and data administrators. Cryptography techniques are in the forefront in providing security to privacy of the data. The need for data privacy is increasing day-by-day. Hence, research in this area is also gaining importance. Data privacy is concerned with confidentiality. According to Reiskind [1], “privacy is about, how you use personal data. It is further concerned with collection of data, the use of data, and the disclosure of the data – to whom, you are giving the data”. People do not like to disclose their private data to others for security reasons. Disclosing the personal data impacts personal life of an individual. In spite knowing that analysis on data can help in discovering hidden patterns and knowledge, data owners are hesitating to share the data because of privacy breach. Homomorphic encryption technique is one the solution to this problem. Homomorphic Encryption (HE) is one of the promising researches in the area of cryptography. HE allows computations on encrypted data, and it yields results which matches with the result of the operations performed on the source data (i.e unencrypted data original data). The purpose of homomorphic encryption is to allow computation on encrypted data. Homomorphic encryption model is raising its significance in the data security and privacy area. In homomorphic encryption, data is encrypted without applying keys (i.e private key and public key). The limitation of public and private key cryptography is that, if the private keys are not kept protected, then security is same as that of password authentication. Along with this, distribution of public keys is not very scalable, as it is cumbersome in large environments. Encryption and decryption is time taking and uses more computer resources. Several techniques have been proposed and available to perform homomorphic encryption like Gentry’s Bootstrap theorem, paillier’s encryption etc [2]. The first fully homomorphic encryption was introduced in the year 2009 [3].

1.1 Types of Homomorphic Encryption:

Researchers perceive homomorphic encryption in three ways based on the number of operations allowed on encrypted data:

1. Fully Homomorphic Encryption (FHE)
2. Some What Homomorphic Encryption (SWHE)
3. Partially Homomorphic Encryption (PHE)

1.1.1 Fully Homomorphic Encryption (FHE)

A cryptographic system that supports computation on encrypted data is known as fully homomorphic encryption (FHE) system [3] [4][5]. Fully Homomorphic Encryption (FHE) allows an unlimited number of operations, unlimited number of times, performed on cipher text, giving similar results as performed on plaintext. Since these systems need not be decrypted, it can run on even un-trusted third party systems. These systems have great scope in cloud computing overcoming the threat of privacy breach.

1.1.2 Some What Homomorphic Encryption (SWHE)

In somewhat homomorphic encryption, no re-encryption is required and allows only some types of operations, limited number of times on encrypted data [2][5].

1.1.3 Partially Homomorphic Encryption (PHE)

Partially Homomorphic Encryption (PHE) allows only one type of operation (either addition or multiplication), unlimited number of times. PHE schemes are in general more efficient than SHE and FHE, mainly because they are homomorphic with respect to only one type of operation [1][5].

Health care industry needs much attention in maintaining the data privacy. Commonly, health care data includes patient medical records, which need to be secured from various attacks. Organizations like hospitals and medical companies majorly focus on data safety and confidentiality. The concept of medical data sharing is on high rise, for gaining business competency and data insights. Health care systems can be best benefited by using homomorphic encryption technologies. This will enable data owners to share medical data for analysis and thus discovering new symptoms and medicines for the wellbeing of the patients. This concept is not limited to health care. It is applicable in all service domains.

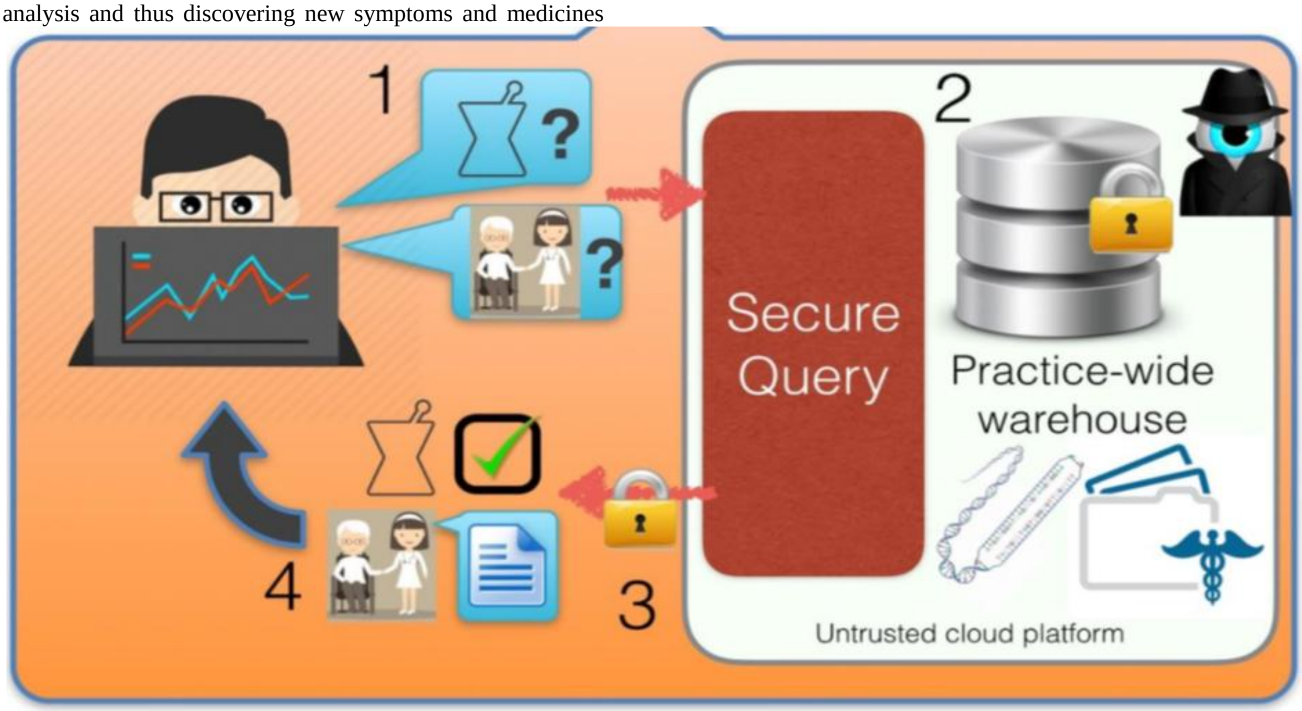

Fig.1 Hospital Analytic workflows over sensitive data using Homomorphic Encryption

(Source: Google Images)

In this paper, the authors present a Somewhat Homomorphic Cryptosystem for privacy preserving sharing of sensitive data using Filter Based Encryption Technique and Particle Swarm Optimization (PSO) algorithm. PSO algorithm has been used to identify the best filter co-efficient, to perform convolution on medical data, thus applying filter based encryption technique.

The basics of PSO optimization is as follows:

2. PARTICLE SWARM OPTIMIZATION

Particle Swarm Optimization was introduced by Kennedy and Eberhart (1995) [6]. It has roots in the simulation of social behaviours using tools and ideas taken from computer graphics and social psychology research. Particle swarm optimization is a population based stochastic optimization technique for the solution of continuous optimization problems. It is inspired by social behaviors in flocks of birds and schools of fish. In particle swarm optimization (PSO), a set of software agents called particles search for good solutions to a given continuous optimization problem. Each particle is a solution of the considered problem and uses its own experience and the experience of neighbor particles to choose how to move in the search space. In practice, in the initialization phase, each particle possesses random initial position and initial velocity. The position of the particle represents a solution of the problem and has therefore a value, given by the objective function. While moving in the search space, particles memorize the position of the best solution they found. At each iteration of the algorithm, each particle moves with a velocity that is a weighted sum of three components: the old velocity, a velocity component that drives the particle towards the location in the search space where it previously found the best solution so far, and a velocity component that drives the particle towards the location in the search space where the neighbor particles found the best solution so far. PSO has been applied to many different problems and is another example of successful artificial/engineering swarm intelligence system.

Part-I narrates the somewhat homomorphic encryption model. A part of this work is available at [7] [8]. PSO finds the best particle to encrypt the original data. Further, in Part-II, the encrypted data usability is tested using predictive modelling techniques classification tree and clustering technique.

3. LITERATURE REVIEW

W. K. Wong et al. [9] have proposed substitution cipher techniques in the encryption of transactional data for outsourcing association rule mining. After identifying the non-trivial threats to a straightforward one-to-one item mapping substitution cipher, they propose a more secure encryption scheme based on a one-to-n item mapping that transforms transactions non-deterministically, yet guarantees correct decryption. They developed an effective and efficient encryption algorithm based on this method. The algorithm performed a single pass over the database and suitable for applications in which data owners send streams of transactions to the service provider. The results showed that the technique was highly secure with a low data transformation cost.

Ling Qiu et al. [10] have proposed an approach for preserving privacy in association rule mining. The main idea is to use keyed Bloom filters to represent transactions as well as data items. Applying filters is one of the encryption techniques. This approach preserves privacy while maintaining the precision of mining results. The trade-off between mining precision and storage requirement was investigated. They also proposed - folding technique to further reduce the storage requirement without sacrificing mining precision and running time.

Jun-Lin Lin and Yung-Wei Cheng [11] investigated the problem of privacy-preserving mining of frequent item sets. They presented a procedure to protect the privacy of data by adding noisy items to each transaction. An algorithm is proposed to reconstruct frequent item sets from these noise-added transactions. The experimental results indicated that the method could achieve a rather high level of accuracy. The method utilizes existing algorithms for frequent item set mining, and thereby takes full advantage of their progress to mine frequent item sets efficiently.

Maneesh Upmanyu, Anoop M. Namboodiri, Kannan Srinathan, and C.V. Jawahar [12] suggest a method for privacy preserving, which overcomes problems of traditional cryptographic protocols such as Secure Multiparty Computation (SMC) and/or homomorphic encryption schemes like Paillier's encryption. In their work, authors focused on paradigm of secret sharing, which allows the data to be divided into multiple shares and processed separately at different servers. With the idea of paradigm of secret sharing, the authors designed secured, cloud computing based solution which has negligible communication overhead compared to SMC and proved that the proposed method is a million times faster than similar SMC based protocols. The encryption usability issues are not addressed in this publication.

Michael Brenner, Henning Perl and Matthew Smith [13] published paper stating the practical applications of homomorphic encryption. The authors stated simple algebraic homomorphic implementation on numeric numbers. The authors aim to present algebraically homomorphic scheme applied on numeric numbers where re-encryption is not required. The authors tried to give proof of correctness. The encrypted data is used for search operation; fuzzy search, soft searches and padding are indicated in this paper. The authors expressed their concern that fully homomorphic implementation is difficult to achieve.

Zvika Brakerski [14] published the fundamentals of Fully Homomorphic Encryption (FHE) like homomorphic functions, definitions, origins of the homomorphic encryptions and highlighted the use of fully encrypted homomorphic encryption. Also, this survey paper gives details of the merits, demerits of the public & private keys, and Gentry's Boot Strap theorem. The authors proposed a few extensions to the Fully Homomorphic Encryption (FHE).

Monique Ogburna at el. [15], published a paper on Homomorphic encryption. This paper described the homomorphic fundamentals, limitations of the homomorphic encryption, and applications (Finance & Medical industry) of the homomorphic encryption. Apart from the basic concepts, the paper implements an algorithm based on homomorphic encryption using patient blood pressure data and provided the proof of concept at the end. The patient blood pressure numbers are encoded into ASCII values and then multiplied by 100. Further the data outsourced to third party for testing.

Iram Ahmad et al [16] explained how Homomorphic Encryption Method can be applied to Cloud Computing. The objective of the paper was to protect the privacy of the data on Cloud Environments. The paper proposes an application of a method to perform the operation on encrypted data without decrypting. The application is tested on raw data and repeated on the encrypted data and finally compared the results of both the computations (i.e with encryption computation and without encryption computation). The whole scenario is illustrated by assuming data is stored on a cloud environment. Because, Cloud environments needs to provide security and privacy preservation.

The review of literature and existing implementation of homomorphic systems clearly states that fully homomorphic encryption is not yet practical, because of significant performance limitations and is too slow for practical applications.

4. PRIVACY PRESERVING BASED ON HOMOMORPHIC MODEL

4.1 Part -1

Before explaining the proposed methodology, the authors present the Conventional privacy methods. One of the techniques for privacy preservation is randomization. It was first introduced by Agarwal and Srikanth [17]. It works on the principle of adding random noise to the original data. For example, raw numeric data multiplied with assumed numeric data. Hence, the data values changes or converted into encrypted formats. Another technique to privacy preserving data mining is to swap the values across the tuples in a relation. Data values are not obscured at all, but data inference problem can be satisfied to some extent. Aggregation is the another method used in privacy preserving data mining, where k-number of records of a data set will be replaced by representative records. The values of representative records are formed by taking the average value of all values. The problem with this method is information loss. One more way to achieve privacy preserving data mining is to suppress all sensitive data.

In paper explains the design and implementation of proposed privacy preserving clustering model for achieving the data privacy of medical data and its usability for data clustering using somewhat homomorphic encryption technique – filter based encryption technique and optimized the results using particle swarm optimization technique - swarm intelligence algorithm.

4.2 Model description

The figure 2.0 shows the implementation flow of Privacy preserving Homomorphic Encryption model.

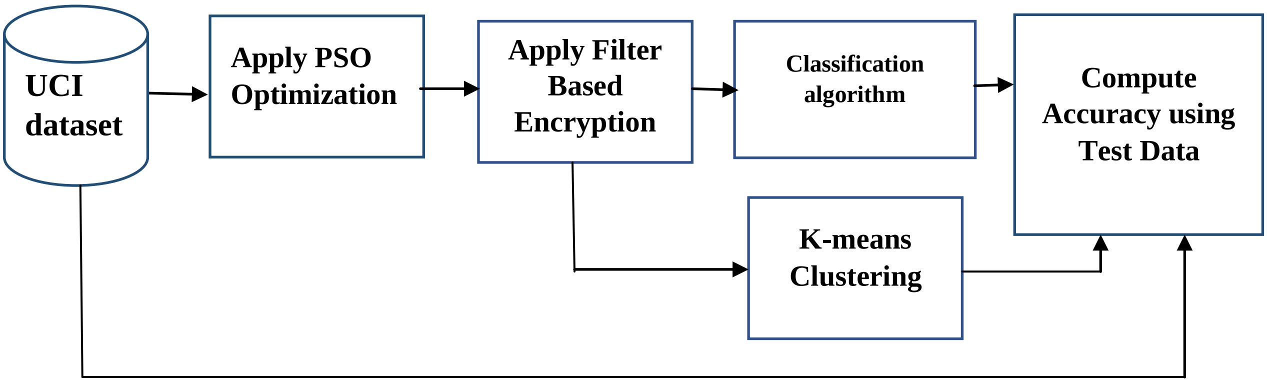

Fig. 2 Privacy Preserving Homomorphic Encryption Model

The following steps (a) to (f) give the implementation details of this work.

a. Data Extraction

b. Data pre-processing

c. Applying particle swarm optimization using clustering techniques

d. Applying cryptographic technique

e. Evaluate usability of the encrypted data using predictive techniques – classification and clustering.

f. Applying Homomorphic operations on encrypted data.

4.2.1 Data Extraction

This step involves with the extraction of data from the external sources. Here the external source is: UCI machine learning repository. The UCI machine learning repository maintains many datasets suitable for machine learning operations (e.g. supervised & unsupervised learning). The researcher extracted Cleveland heart medical data set for optimization and predictive operations. Cleveland data set has class labels indicating angiographic (heart) disease status.

4.2.2 Data Pre-Processing

It identifies missing vales and replaces them with mean value, normalizes the input data, randomizes the input data and samples the input into: train data, train class data, test data, test class data. Train data is 80% and test data is 20% [8] [9]. The following code snippet shows the data separation technique followed in this work:

- train_feature_data=randomized_feature_data (1: round (size(feature_data,1)*0.8),:);
- train_class_data=randomized_class_data (1:round(size(feature_data, 1)*0.8),:);
- test_feature_data=randomized_feature_data(round(size(feature_data, 1)*0.8)+1:end,:);
- test_class_data=randomized_class_data(round(size(feature_data,1)*0.8)+1:end,:);

4.2.3 Applying Particle Swarm Optimization (Swarm-Kmeans)

It performs PSO optimization considering 20 particles of 25 dimensions each. The PSO optimization filters the best fit particle (best filter co-efficient) based on input data for encryption purpose. The best fit particle is reshaped to produce a filter coefficient matrix of size 5 x 5, which is big enough to conceal the original values.

4.2.4 Applying Cryptographic Technique – (Convolution method)

The obtained 5 x 5 filter coefficient matrix is convoluted with the medical data to obtain encrypted medical data. 2-D convolution has been performed. The current work encrypts feature data (i.e input data without class data) using filter based coefficient matrix. For example, con=conv2 (D, E) computes the two-dimensional convolution of matrices D and E and stores the result into 'con' variable. D is the input data, E is a two-dimensional finite impulse response (FIR) filter and 'con' contains convoluted matrix, i.e. encrypted medical data. Homomorphic operations can be performed on this encrypted data.

4.2.5 Evaluate usability of the data using predictive technique

The objective of this step is to evaluate the usability of the data after encryption process and to check whether homomorphic encryption preserves the inherent relations between the underlying attributes of the data. To achieve the objective, predictive analytics using classification and clustering have been applied on the data. Details of predictive modeling is described in section part-II.

4.2.6 Applying Homomorphic operations on encrypted data

To evaluate the homomorphic encryption model, some of the homomorphic operations have been performed on the encrypted data and then predictive accuracy using classification has been observed. The operations performed on encrypted data are (i) Converted encrypted data to absolute values (ii) Rounded off the encrypted data to 3 decimal places, from 8 decimal places (iii) Multiplied the encrypted data with 10000. The observed results have been tabulated in Table 2. It has been observed that there is no variation in prediction accuracy even after applying different operations on encrypted data. This shows that the model preserves the relationships between the attributes, even after applying operations on encrypted data.

4.3 Predictive Modelling

Predictive modelling algorithms perform predictive operations on the new data based on the relations in the old data [18]. Predictive modelling is often used in the machine learning area. There are two types of learning algorithms: supervised learning algorithm and un-supervised learning algorithm [19] [20] [21]. Supervised algorithm requires training data with target value to create predictive model. Further, predictive model accuracy is tested using test data. Often test data does not contain target value, which will be predicted using predictive model.

The researcher applies classification algorithm (ID3) for tree model generation. Subsequently, the model performance is evaluated using test data. The same process is applied on the encrypted Cleveland data. At the end, the predictive accuracy is evaluated between the un-encrypted Cleveland test data and encrypted Cleveland test data. The observed values are given in the Results & Evaluation section.

In this work, K-means clustering has also been applied on encrypted Cleveland dataset to compute clustering accuracy. The observed values are given in the Results & Evaluation section. The Privacy Preserving Homomorphic Encryption Model has been evaluated using both classification and clustering. K-means represent un-supervised learning technique and classification tree represents supervised learning technique. The following section discusses the results & evaluation.

4.4 Results & Evaluation

This section explains results in two parts: Predictive results using classification tree (ID3) and results using K-means clustering.

4.4.1 Predictive results using classification tree:

The raw data is divided into sets: training (80%) and test (20%). To control the computational nuisance, repeated sampling method is chosen for data selection with repetition count 10. Information gain chosen as splitting criteria. In order to get better estimation, 10-fold cross validation method is employed. The final model is chosen based on the computational parameter (CP) with least error. The best fit model is applied on the test data to predict the target value. Finally, the predictive power is computed using confusion matrix.

The following fig 2 shows the code snippet of the predictive modelling implementation using R programming tools:

```
# Block1: Training the Decision Tree classifier with criterion as information gain.
Step1: trctrl <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
Ste2: set.seed(3333)
Ste3: f=factor(training$F14)
# Block2: Generate the predictive model
Step4: dtree_fit = train(training, f, method = "rpart",parms = list(split = "information"),trControl=trctrl,tuneLength
= 10)
step5:FirstPredict=predict(dtree_fit, newdata=testing)
step6: x=confusionMatrix (FirstPredict, testing$F14)
```

The fig 3.0 shows R programming code showing data partitioning and predictive model. This experiment follows 10-fold method for best tree model selection. Generally, tree model is evaluated using error and Complexity Parameter (CP). Since, it is 10-fold method, 10 different tree models are generated each consisting CP and error. Out of the 10 CP values, the best CP value with least error tree is chosen as the final tree model (called dtree_fit). Further, dtree_fit model is applied on test data to generate the prediction values (i.e 0,1,2,3 and 4). The step 6 of Block 2 generates confusion matrix, which compares predicted values and actual values of test data. Based on the comparison, the confusion matrix finds prediction accuracy. The experiment (i.e Block1 and Block2 code) applied on raw data followed by encrypted data. The following table shows the different accuracy measures obtained.

Table 1: Prediction Accuracy Measures(Raw Vs Encrypted)

S.No	Data	Prediction Accuracy	Confidence	P-Value	Kappa Value	CP Value	Tree Accuracy
1	**Raw Data**	0.5167	95%	0.3489	0.2731032	0.03960905	0.5963361
2	**Encrypted Data**	0.6667	95%	0.5603	0.0238392	0.02521008	0.4581594

Table 2: Prediction Accuracy Measures after applying homomorphic operations on encrypted data

S.No	Data	Prediction Accuracy	Confidence	P-Value	Kappa Value	CP Value	Tree Accuracy
1	**Encrypted Data**	**0.6667**	**95%**	**0.5603**	**0.0238392**	**0.02521008**	**0.4581594**
2	**Encrypted Data with Absolute values**	**0.6667**	**95%**	**0.5603**	**0.000132645**	**0.03781513**	**0.5051098**

3	**Encrypted Data rounded off to 3 decimal values**	**0.6667**	**95%**	**0.5603**	**0.005529081**	**0.03781513**	**0.5050543**
4	**Encrypted data multiplied by 10000**	**0.6667**	**95%**	**0.5603**	**0.001326446**	**0.03781513**	**0.5051098**

The first row of the Table 1 shows measures based on raw data; prediction accuracy, confidence interval, P-values, Kappa value, CP value, and tree accuracy. Second row shows measures based on encrypted data. Initially, the encrypted data has values with 8 decimal places. These values been scaled to; encrypted data with absolute values, encrypted data rounded off to 3 decimal values and encrypted data multiplied by 10000. The prediction accuracy based on raw data is 0.5167 and tree accuracy is 0.5963361. In all the remaining encrypted cases (Table 2 - row number 1 to 4) the prediction accuracy (i.e. 0.6667) and tree accuracy (i.e. 0.5051098) are consistent. Tree accuracy is same up to first decimal place in all the cases, whereas prediction accuracy between raw data and encrypted data varies by degree of 0.1500. It is significant at 95% confidence interval. However, the P-values are above 50%.

4.4.2 Results using K-means clustering

The encrypted Cleveland data is passed to K-means clustering technique to check the accuracy of the clustering on encrypted data. Clustering accuracy is computed by comparing the obtained class by clustering with the class label of the original data set. Computation of clustering accuracy is repeated for clusters ranging from 2 to 10. Clustering accuracy is computed using average precision purity method [22]. The results of the accuracy are shown below:

Table 3. Clustering accuracy on encrypted data

No of clusters	K-means: Clustering Accuracy
2	0.513017
3	0.64087
4	0.55357
5	0.61261
6	0.59038
7	0.57393
8	0.57924
9	0.5821
10	0.61893

The table 1 and table 3 shows accuracy results using classification tree and clustering. Though the two techniques are not same, the accuracies are measured on a scale of 0 to 1. Many of the clustering accuracy results are close to classification accuracy of raw and encrypted data.

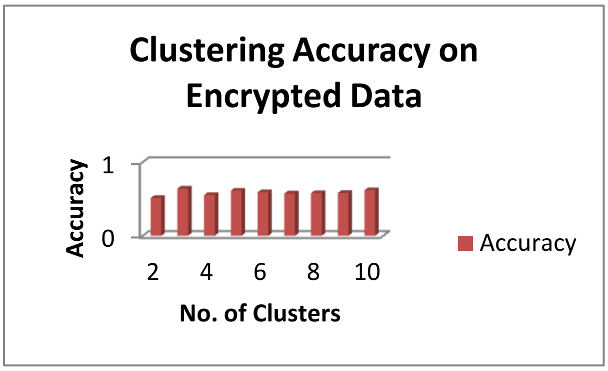

Fig .3 Accuracy Levels with Respect to Clusters

The fig 3.0 represents the graphical representation of the accuracy achieved on encrypted data through clustering.

5. CONCLUSION

The present paper highlights homomorphic encryption using PSO optimization and filter based encryption technique. Homomorphic encryption overrides the limitations of private and public keys. It is new trend in the cryptographic area. The present work shows PSO optimization along with encryption. The usability and accuracy of encrypted data is tested using predictive classification technique and clustering technique. The observed results are satisfactory. Data privacy and utility needs of healthcare organizations are growing rapidly. Homomorphic encryption may provide a novel solution to some of the tradeoffs, at minimal cost. While going through literature review; the authors of this paper have come to know that standardization of Homomorphic encryption is in progress. Hope, progress will be seen in this area of research.

6. ACKNOWLEDGMENTS

The Authors wish to thank Abacus Institute of Computer Applications, Research Centre, Savitribhai Phule Pune University, Hadapsar and Kirloskar Institute of Advanced Management Studies for providing the opportunity to publish this content.

7. REFERENCES

[1] Michael Minelli et al., Big data big analytics Emerging Business Intelligence and Analytic Trends for today's businesses, Wiley publications, ISBN 978-1-118-14760-3 (cloth); ISBN 978-1-118-22583-7 (ebk), 2013.

[2] Craig Stuntz (2010-03-18). "What is Homomorphic Encryption, and Why Should I Care?"

[3] Craig Gentry, A Fully Homomorphic Encryption Scheme, (Dissertation), 2009. Link available at https://crypto.stanford.edu/craig/craig-thesis.pdf

[4] Craig Gentry. Fully homomorphic encryption using ideal lattices. In Proceedings of the 41st ACM Symposium on Theory of Computing – STOC 2009, pages 169–178. ACM, 2009.

[5] A Survey on Homomorphic Encryption Schemes: Theory and Implementation ABBAS ACAR, HIDAYET AKSU,

and A. SELCUK ULUAGAC, Florida International University MAURO CONTI, University of Padua. Online available at https://arxiv.org/abs/1704.03578.

[6] Kennedy, J.; Eberhart, R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks. IV. pp. 1942–1948.

[7] Asha kiran, Manimala Puri, Srinivasa Suresh, "A Model for Preserving Data Privacy Using Optimization Technique," International Conference on Electrical, Electronics, Computers, Communication, Mechanical and Computing (EECCMC)-2018, Jan 2018, IEEE, Vellore, Tamil Nadu, ISBN: CFP18037-PRI: 978-1-5386-4303-7.

[8] Asha kiran, Manimala Puri, Srinivasa Suresh, "Application of PSO Optimization Technique on Medical Data to Uphold Data Privacy", International Conference on Science, Technology, Engineering and Management, 1 st and 2 nd August 2017, Dubai, ISBN: 978-93-86831-14-9.

[9] Hao-Miao Yang, Qi Xia, Xiao-fen Wang, and Dian-hua Tang. 2012. A new somewhat homomorphic encryption scheme over integers. In Computer Distributed Control and Intelligent Environmental Monitoring (CDCIEM), 2012 International Conference on. IEEE, 61–64. Andrew Chi-Chih Yao. 1982. Protocols for secure computations.

[10] Ling Qin et al., Preserving Privacy with Association rule mining with bloom filters, J Intell Inf Syst (2007) 29:253–278 DOI 10.1007/s10844-006-0018-8. Springer.

[11] Jun-Lin Lin and Yung-Wei Cheng Privacy Preserving itemset mining through noisy items, Journal of Expert Systems with Applications, Volume 36, issue 3, April 2009. Pages 5711-5717.

[12] Maneesh Upmanyu,Anoop M. et al, "Efficient Privacy Preserving K-Means Clustering", H. Chen et al. (Eds.): PAISI 2010, LNCS 6122, pp. 154–166, 2010. c Springer-Verlag Berlin Heidelberg.

[13] Michael Brenner, Henning Perl and Matthew Smith, Practical Applications of Homomorphic Encryption, SECRYPT, 2012 International Conference on Security and Cryptography

[14] Zvika Brakersky, Fundamentals of Fully Homomorphic Encryption – A Survey, Electronic Colloquium on Computational Complexity, Report No. 125 (2018).

[15] Monique Ogburna*, Claude Turnerb, Pushkar DahalcComplex Adaptive Systems, Publication 3 Cihan H. Dagli, ELSEVIER, Editor in Chief Conference Organized by Missouri University of Science and Technology 2013- Baltimore, MD Homomorphic Encryption. Procedia (2013).

[16] Iram Ahmad 1 and Archana Khandekar, Homomorphic Encryption Method Applied to Cloud Computing, International Journal of Information & Computation Technology. ISSN 0974-2239 Volume 4, Number 15 (2014), pp. 1519-1530 © International Research Publications House http://www. irphouse.com.

[17] Charu C. Aggarwa l III Philip S. YA Survey of Randomization Methods for Privacy-Preserving Data Mining. Springer link available at https://link.springer.com/chapter/10.1007/978-0-387-70992-5_6

[18] "Predictive modelling available at https://www.sas.com/ko_kr/insights/analytics/predictive-modeling-techniques.html

[19] Estivill-Castro, Vladimir (20 June 2002). "Why so many clustering algorithms – A Position Paper". ACM SIGKDD Explorations Newsletter. 4 (1): 65–75. doi:10.1145/568574.568575.

[20] Everitt, Brian (2011). Cluster analysis. Chichester, West Sussex, U.K: Wiley.ISBN 9780470749913.

[21] Jewie Hann et al, Data Mining Techniques and Concepts, The Morgan Kaufmann Series in Data Management Systems, Jim Gray, Series Editor Morgan Kaufmann Publishers, August 2000. 550 pages. ISBN 1-55860-489-8

[22] Asha kiran, Manimala Puri, Srinivasa Suresh , "Average Precision Purity Algorithm for Evaluating Clustering Accuracy", Journal of Advance Research in Dynamical & Control Systems, Institute of Advanced Scientific Research, USA, ISSN 1943-023X Vol. 10, 02-Special Issue, 2018.

[23] J. Kim, et al. Encrypting Controller using Fully Homomorphic Encryption for Security of Cyber-Physical Systems, IFAC, Vol. 49, Issue. 22, pp.175-180, 2016.

[24] A Survey on Homomorphic Encryption Schemes: Theory and ImplementationABBAS ACAR, HIDAYET AKSU, and A. SELCUK ULUAGAC, Florida International University MAURO CONTI, University of Padua,

[25] In FOCS, Vol. 82. 160–164. Xiaojun Zhang, Chunxiang Xu, ChunhuaJin, Run Xie, and Jining Zhao. 2014. Efficient fully homomorphic encryption from RLWE with an extension to a threshold encryption scheme. Future Generation Computer Systems 36 (2014), 180–186.

[26] Zhenfei Zhang. 2014. Revisiting fully homomorphic encryption schemes and their cryptographic primitives. (2014).

[27] Yuliang Zheng, Tsutomu Matsumoto, and Hideki Imai. Cryptographic Applications of th-Residuosity Problem with an Odd Integer.