CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Machine Learning based Recommendation of Ensembled Cryptographic Algorithms for Plaintext

by

Samreen Saeed

A thesis submitted in partial fulfillment for the degree of Master of Science

in the

Faculty of Computing
Department of Computer Science

2021

This effort is dedicated to ***ALLAH ALMIGHTY*** *who is the creator of all the worlds and everything within there.*

Then to my respected supervisor Dr. Qamar Mehmood for his support and guidance throughout the research.

My beloved parents for their love, care and support. I also dedicate this to my Partner, My Siblings and my small Circle of friends for their support and encouragement.

CERTIFICATE OF APPROVAL

Machine Learning based Recommendation of Ensembled Cryptographic Algorithms for Plaintext

by

Samreen Saeed

(MCS173062)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Mohammad Imran	Air University, Islamabad
(b)	Internal Examiner	Dr. Abdul Basit Siddiqui	CUST, Islamabad
(c)	Supervisor	Dr. Qamar Mehmood	CUST, Islamabad

Dr. Qamar Mehmood
Thesis Supervisor
December, 2021

Dr. Nayyer Masood
Head
Dept. of Computer Science
December, 2021

Dr. M. Abdul Qadir
Dean
Faculty of Computing
December, 2021

Author's Declaration

I, **Samreen Saeed** hereby state that my MS thesis titled "**Machine Learning based Recommendation of Ensembled Cryptographic Algorithms for Plaintext**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MS Degree.

(Samreen Saeed)
Registration No: MCS173062

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "**Machine Learning based Recommendation of Ensembled Cryptographic Algorithms for Plaintext**" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS Degree, the University reserves the right to withdraw/revoke my MS degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

(Samreen Saeed)
Registration No: MCS173062

Acknowledgement

First of all, I give my gratitude to **ALLAH ALMIGHTY**, Who is the creator of all the worlds and all the creatures therein for giving me the courage and strength to carry out this research work. Without the will and help of **ALLAH ALMIGHTY**, I would not be able to write a single word. After that thank you to my supervisor, **Dr. Qamar Mehmood**, for the patience, guidance, and support he has provided to me throughout the research. I have benefited greatly from your wealth of knowledge and precise editing. I am extremely thankful that you took me under your supervision and continued to have faith in me. Nobody is beloved to me more than my Mother and Father whose love and guidance is with me since the day I was born. So I would like to thank my beloved parents for their love, care, support, and guidance throughout the journey. A special thanks to my partner for his support and encouragement to complete this Master's degree. I also thank my small circle of friends for their encouragement and motivation that helped me to complete this research and degree. I pray to **ALLAH ALMIGHTY** that He showers His blessing upon me and gives me success in this world and hereafter as well.

(Samreen Saeed)

Abstract

Cryptography is the domain of information security that concerns converting the data into a specific form so that it is only readable by the sender or receiver of the data. There has been a rapid and significant amount of work done in the domain of cryptography. Researchers have combined symmetric and asymmetric cryptographic algorithms to create a more strong cryptography algorithm. However, there is a lack of technique that suggests the combination of different algorithms based on the plaintext content and its type. We have proposed a technique that suggests an ensembled cryptographic algorithm using a machine learning model based on the type and content of the plaintext. Supervised machine learning is used to experiment on a dataset created by a user survey. We have trained multiple machine learning classifiers and have compared their results. We have used SVM, Random forest, Decision tree and Naive bayes algorithm to carry out our experiment. The experiment results showed that the random forest algorithm gave the best performance with 88.4% accuracy.

Contents

List of Figures

List of Tables

Abbreviations

AES	Advanced Encryption Standard
DES	Data Encryption Standard
ECC	Elliptical Curve Cryptography
FP	False Positive
FN	False Negative
ML	Machine Learning
NB	Naive Bayes
RF	Random Forest
RSA	Rivest Shamir Adleman
SVM	Support Vector Mchimes
TP	True Positive
TN	True Negative

Chapter 1

Introduction

This chapter will introduce the domain of the research. This chapter has sections that explain the background of the proposed system. The concept of cryptography and ensembled cryptography has been explained in this chapter. This chapter also discusses the proposed technique and its methodology in which the combination of the symmetric and asymmetric algorithms has been recommended based on plaintext type.

1.1 Domain Introduction

In this section, we have discussed cryptography and its types in detail. We have also explained ensembled cryptography in this section.

Cryptography is the domain of information security that concerns converting the message into a specific form so that it is only readable by the sender or receiver of the data. Cryptography provides secure communication in the existence of malicious third parties. Cryptography has two primary objectives: Confidentiality and Integrity.These objectives of cryptography ensure a secure communication between sender and receiver. These objectives belong to the **CIA triad**.

Confidentiality means that only the receiver of the message should be able to read it and the message should be unreadable by any malicious third party.

Integrity means that the message should be secure enough that even some third party should not be able to change it while transmission.

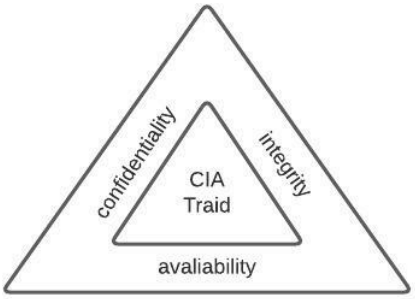


FIGURE 1.1: CIA Traid

Taking the original message called plaintext, and converting it into an unreadable form called ciphertext, is called encryption. The key allows the receiver of the message to convert the message into its original form again. This process is called decryption. The keys basically lock or unlock the algorithms, allowing the encryption and decryption process to happen. There are two types of cryptography algorithms available: symmetric and asymmetric. Both types are used to perform encryption [1], but their process is slightly different.

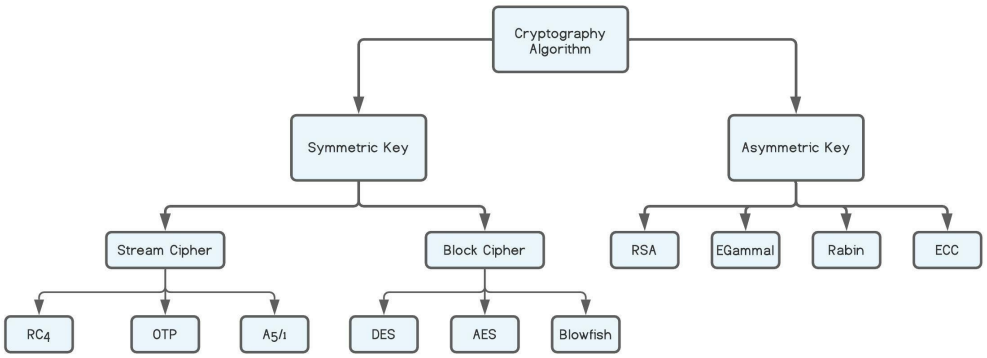


FIGURE 1.2: Types of Cryptography

Symmetric algorithms uses a same key for encryption and decryption process. The key should be same and known to both the sender and receiver for a successful and secure transmission [1]. They are further divided into different types as well.

Asymmetric algorithms use two keys for the encryption and decryption process: a public key and a private key. The sender uses a public key to encrypt the

plaintext while the receiver uses the private key to decrypt the ciphertext into plaintext [1]. Asymmetric algorithms are further divided into two further types.

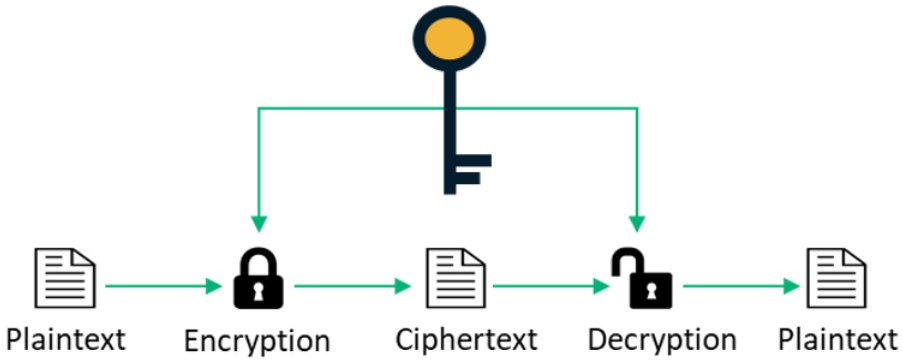


FIGURE 1.3: Symmetric algorithms [2]

Block cipher algorithms perform encryption on a group of bits called blocks. These algorithms are mostly used to encrypt the offline type of data or the data which is at rest[1].

Stream cipher algorithms perform encryption bit by bit or a byte of plaintext at a time. These algorithms are mostly used to encrypt the online nature of data or the data which is travelling over some sort of network [1].

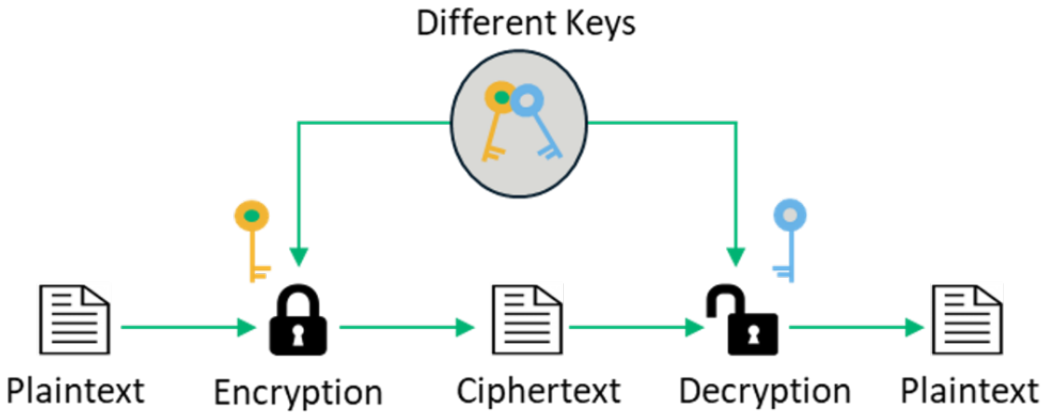


FIGURE 1.4: Asymmetric algorithms [2]

Other than these two types, cryptography has another type, which is called **Hash functions**. The hash functions are one-way functions that convert a string into a fixed-length string. Hash functions convert an input of a random length into some output of random length which is either compressed or unreadable. Hashing functions are not recoverable, which means that the output from a hash function

cannot be reversed into the original format. Some of the well-known hashing functions are MD5, SHA1, and SHA2. etc. Figure 1.6 explains the working of hashing functions. **A**n Ensembled Cryptographic algorithm is a term used when multiple cryptographic algorithms are combined together to use their strength together [3].

Different combinations of Symmetric and Asymmetric algorithms are merged together to form the ensembled cryptographic algorithms. This creates a more stronger cryptographic algorithm which is hard to break by the third party.

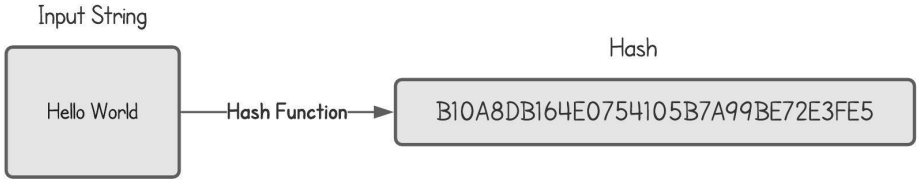


FIGURE 1.5: Hashing Functions

1.2 Rationale of Research

According to our knowledge, there has been a significant amount of research done for creating ensembled cryptographic algorithms. Researchers have come up with different techniques which combine cryptography algorithms to use their strength together and create a more strong cryptography algorithm [3]. Those algorithms have seemed to perform well. But, different parameters have been considered while implementation of ensembled cryptographic algorithms such as key size and block size.

The data type is an important parameter for cryptography, which is neglected. Type of plaintext can be offline or online, which means that the data which is at rest or not connected to the network is the offline type and the data which is traveling over the network is the online type. The combination of cryptography algorithms changes when the type of plaintext changes. This is not considered in

the previous researches. Our proposed technique considers this neglected parameter and deals with the data type for suggesting the combination of cryptographic algorithms. This technique will recommend the combination of different algorithms based on the type of plaintext. Using this technique, different systems can be implemented, which will apply different combinations of cryptographic algorithms considering the data's nature. This technique will use machine learning to recommend the combination of cryptographic algorithm based on type of text.

1.3 Sufficiency of topic to Qualify as MS Thesis

Previously, various parameters have been considered while combining different cryptographic algorithms such as key size and block size. Data type is an important parameter while combining the algorithms together [3], which is neglected in the research. Mostly block cipher algorithms are used for the offline type of data or the data which is at rest [1]. Likewise, for an online type of data or the data which is travelling as bytes is encrypted by stream cipher algorithms [1].
Our proposed technique deals with the data type for suggesting the combination of cryptographic algorithm using machine learning. This technique will recommend the combinations based on the data type and the content of plaintext.

1.4 Problem Statement

Currently, there is a lack of technique that recommends the combination of different symmetric and asymmetric algorithms by considering the plaintext type and its content using a machine learning technique.

1.5 Research Questions

1. Can we suggest the combination of cryptographic algorithms considering the plaintext type and its contents using machine learning?

2. How well machine learning model can take part in recommending the combination of cryptographic algorithms

1.6 Proposed Solution

Our proposed solution considers the parameter (plaintext nature) that has been neglected in previous researches. The proposed technique recommends the combination of cryptographic algorithms based on plaintext nature. We have used machine learning to implement this technique. The machine learning model has been trained using plaintext and its type. Four machine learning classifiers have been trained on the training data and tested on the test data provided. Classifiers names are SVM, Random Forest, Decision Tree, and Naive Bayes. The trained classifiers recommend the ensembled cryptographic algorithm by training themselves on the train data created with the help of subject experts.

1.7 Research Objectives

To develop a system which will recommend the combination of cryptographic algorithms based on plaintext nature. To achieve this, we are going to use machine learning technique and will implement a machine learning model.
Machine learning model performance should not be very much affected and the model should train and test itself in a good amount of time.

1.8 Research Contribution

1. Previous researchers have implemented many techniques which combine different cryptographic algorithms together based on different parameters like block size and key size. Some other parameters such as algorithm performance and time taken to execute are also considered. But they have neglected an important parameter which is plaintext type or nature.

2. Our technique considers the neglected parameter which is plaintext type. This technique recommends the combination of cryptographic algorithms considering the plaintext nature using machine learning.

1.9 Research Method

This section discusses the research method we have used for our technique. We have used an experimental methodology where we experiment to test our hypothesis. This technique uses the machine learning technique to recommend the combination of different algorithms considering the plaintext type. The dataset has been created by user survey with the help of subject experts.

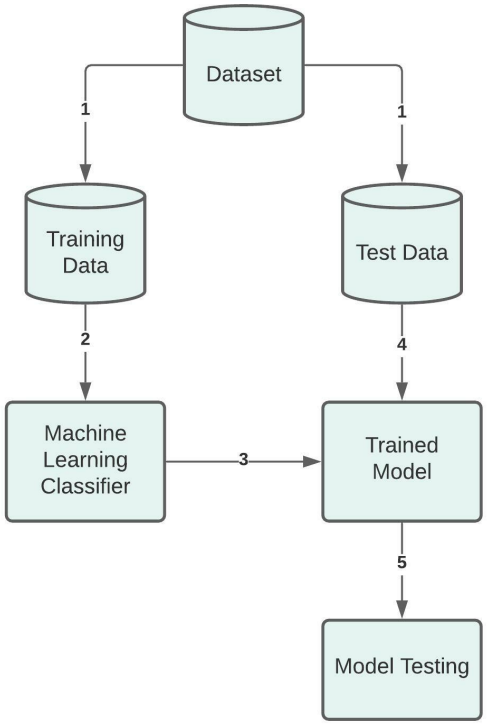


FIGURE 1.6: Research Methodology

1.9.1 Research Method Steps

In this section, we are briefly going to discuss the steps we have performed in our research. We have performed the below set of steps to achieve our goal. Steps are

also shown in graphical format in figure 1.6

1. Collection of the raw data from different online and offline sources in order to create the dataset.

2. After collection of the raw data, distribute the raw data to 4 to 5 subject experts to perform the user survey on data. The subject experts will suggest the combination of cryptography algorithms on each of the data rows. Subject experts will belong to the domain of information security. After taking the suggestions from all the subject experts, we will choose the most recommended combination of cryptography algorithm to create the final dataset for each row.

3. Perform the data preprocessing steps on the data such as cleaning the data, Transformation of the data, Removing the unnecessary fields and data.

4. When the dataset is created and cleaned, divide the data into training and test data. We have used 80% data for training purposes and 20% data for testing purposes.

5. Train the machine learning classifiers on the training data. The machine learning classifiers which we have selected for performing the experiment are discussed in section 3.4

6. Test the machine learning classifiers on the test data and compare their results.

7. The classifier having the best results according to the evaluation metrics we selected is recommended for our technique. The evaluation metrics we selected for our machine learning classifiers evaluation are discussed in section 4.1

1.10 Organization of Thesis

Chapter 2 discusses the literature review related to our proposed technique. It has 4 sections. Section 1 discusses different techniques which have combined multiple

algorithms together to form a more stronger algorithm for cryptography. Next section does the comparative analysis of the discusses techniques. Section 3 discusses the research gaps we identified in the studied techniques and section 4 summarizes the entire chapter.

In chapter 3, the research method of the proposed technique is discussed. It has 8 sections and in these sections, the entire research method is explained in detail. Chapter 4 evaluates the results of the experiment performed.

In chapter 5, we have concluded the thesis, discussed the limitations and considered the future work that can be done.

Chapter 2

Literature Review

There has been a significant amount of work done in the domain of ensembled cryptography by the research community. Researchers have proposed different combinations of symmetric and asymmetric cryptographic algorithms and they have shown to be stronger than a single cryptography algorithm.
In this chapter, we have discussed the work done in cryptography where researchers have implemented the combination of cryptographic algorithms based on several parameters. We have also done the comparative analysis of those techniques and have identified the research gaps in those techniques.

2.1 Survey of Existing Techniques

When we combine multiple cryptography algorithms together, they make a more strong cryptography algorithm that is hard to break by the third party. Cryptography has a very significant amount of research done. Researchers have come up with techniques and have analyzed their performances and security strengths together [4]. They have used several parameters while combining the cryptographic algorithms such as block size and key size. Plaintext type is another important parameter that plays an important role while selecting the cryptographic algorithm or merging multiple algorithms together [1]. In our literature review, we

focus on that are there any techniques available which use plaintext type to make the combination of cryptography algorithm together.

Salama Abdelminaam et al. [5] proposed multiple algorithms by combining different cryptographic algorithms and compare their performances and security strengths with each other. They have used five ensembled encryption methods and points out the strength and weakness of each algorithm. However, this technique has not considered the plaintext nature of their hybrid cryptography algorithms.

M. Harini et al. [6] have presented a technique in which they have combined three cryptography algorithms to ensure the confidentiality,integrity and availability of the data. The algorithms which they have merged to use their strength are AES symmetric algorithm, RSA asymmetric algorithm and MD5 hashing algorithm. They have considered plaintext data for their research. Their results showed improved security of the plaintext. However, they have not considered the plaintext type while designing their algorithm.

Vivek Kapoor et al. [7] have proposed a highly secure technique that combines three cryptography algorithms together. Researchers have used RSA, DES and SHA1 together to present a more strong algorithm. The implementation of this technique is done using JAVA programming language. The evaluation of this technique is done by the space and time complexity. The space and time complexity of this algorithm is compared with the traditional RSA algorithm. They found proposed technique more secure and efficient to generate a ciphertext. In this technique, we did not find any parameter they considered designing the algorithm.

Chitra Biswas et al. [8] have proposed a new cryptography algorithm by using AES and RSA algorithms together. The symmetric key that is being used for encryption process is also encrypted in this technique. This makes sure of better security. Researchers have also created a digital signature for a message by encrypting the hash value of the message. The purpose of the digital signature is to ensure integrity checks upon receiving end. The encrypted message, its key and the signature are merged to form a complete message.The purpose of the digital signature is to ensure integrity checks upon receiving end. The encrypted message, its key and the signature are merged to form a complete message. At the end,

the entire message is being secured using steganography. No parameter has been considered for this technique, such as block size, key size or plaintext type, for the creation of algorithm.

Binay Kumar et al. [3] proposed a technique which applies the block cipher and stream cipher algorithms based on content redundancy found in a plaintext or a text file. This technique comprises two phases: the first phase removes the redundancy of block of texts and references in a file and results in a file whose size is reduced. In the second phase, block cipher and stream cipher algorithms are applied on the text blocks and references. After applying the algorithms, their results are concatenated together in order to get a base64 string. The algorithms are applied on unique blocks to they have considered block size while encryption. However, one of the most important parameter which is plaintext type, is missing in this research.

Li Zhang et al. [9] have presented an ensembled technique for encryption and decryption. They have used the 3DES algorithm for plaintext encryption and they have used the RSA algorithm for encrypting the keys. In this paper first, sender uses the DES algorithm for encrypting the plaintext with the symmetric key and then encrypts the symmetric key using the RSA algorithm, after encrypting the plaintext and key sender transfer this to the network for receiver, after receiving the cipher-text information receiver decrypt the key with its own key to get the DES key and then decrypts the ciphertext using the key and gets the plaintext. The entire system is developed using JAVA programming language. The system results in being simple, effective and has good security but one of the most important parameter which is plaintext nature, has not been considered while developing this system. They have only suggested one new cryptography algorithm.

Shashikant Kuswaha et al. [4] have accomplished the goal by the merger of two algorithms called AES and DES. The input type being supported is text, images, and video files. The input being converted to 128-bit plain-text. In this encryption approach, after that, 128 bits are divided into two blocks of 64 bits each and are given to DES for encryption.
The results from DES are again combined as 128 bit ciphertext and are given

to AES for further encryption. The decryption process is reversed as encryption process. Block size of data is considered in this approach.

Wang Tianfu et al. [10] have come up with a technique of cryptography, which combines two cryptography algorithms. The cryptography algorithms which are being combined are AES and DES. They have designed the system in VC++.

Yasmin Alkady et al. [11] presented a technique which ensures the confidentiality, integrity and availability of the plaintext by combining two algorithms together. ECC and AES are combined to provide a more strong cryptography algorithm. Their results showed that the proposed technique gives excellent results in terms of size of ciphertext, computation time and battery consumption of WSN networks. They have made this system especially for wireless sensor networks. Their results also seemed to be robust in case of image encryption.

Raed Abu Zitar et al. [12] have introduced a method for text encryption in this paper. Authors created a random number generation function that generates sequences of signed random numbers that rely on both plaintext and key. The random numbers support the function of four random operations: random mutation, random cyclic shifting, random permutation, and dirty symbol random insertion. These operations guarantee data security by steadily melting the statistical structure of plaintext and relationships to a key.

Ting Liu et al. [13] have examined the security of medical images in IoT by employing an innovative cryptographic model with optimization procedures. This technique is specifically designed for IoT-based systems which are taking part in medical. For the most part, the patient data gets stored as a cloud server in the hospital due to which the security is a vital state.
Estimation-based Dynamic Encryption and Authentication (SEDEA) scheme is introduced to ensure secure communication between the Control Center (CC) and Remote Terminal Units (RTUs) in the Smart Grid. The general idea of SEDEA is the observed and estimated power system states are employed as a pair of common secrets at RTUs and CC to update the encryption key automatically and synchronously.

Diaa Salama AbdElminaam et al. [14] have presented a technique which creates an ensembled cryptography algorithm by merging two algorithms together. The algorithms that have been merged are AES and Blowfish. First, the 128 bit text block is divided into two blocks of 64 bit. One block is encrypted by using AES and the other block is encrypted by Blowfish. Results of both the encryptions are merged.The key of the algorithms is hashed by MD5 algorithm. In the decryption phase, again ciphertext is divided into two blocks and one block is decrypted by using AES and the MD5 hashed cipher key. Similarly, the other block is decrypted by using Blowfish with the hashed key. The security of the plaintext and the key is increased in this process. The block size is considered in this approach while performing the encryption process.

Ashish Sharma et al. [15] hHave come up with a technique that combines the strengths of RSA and DES algorithms to create a more strong cryptography algorithm. They have specially created this system for MANETS. The data that is to be sent over a mobile ad hoc network is first encrypted by using DES and the encrypted data by DES is further on passed to AES for encryption. AES generates a more strong encrypted data and after that the data is ready to travel over the network. After the receiver mobile device has received the data. It decrypts the data in the reverse format, in which the data was encrypted.

Nishtha Mathur et al. [16] proposed a technique in which a cryptography algorithm is designed in such a way that AES algorithm is used to encrypt the plaintext and ECC cryptography is used to encrypt the AES key. To further improve the security of the system, the AES algorithm is improved to have the key size of 192 bit and 12 rounds of algorithm. A basic AES has 128 bits and 10 rounds of algorithm. The parameters they have considered are key size and no. of iterations of the AES algorithm. The plaintext nature is not found in this research as well.

Babitha.M.P et al. [17] have proposed a research technique which first analyzes the security issues that can occur inside a cloud computing environment and then later on they have proposed a cryptography algorithm that can improve the security of data inside a cloud computing environment. They have not created any new cryptography algorithm. Instead of that, they have used a 128 bit AES

algorithm to improve the confidentiality, integrity and availability of the cloud computing environment. They evaluated their performance based on delay. In their results, they analyzed that when they increased the file size, there was a tremendous increase in delay. They evaluated their performance based on delay. In their results, they analyzed that when they increased the file size, there was a tremendous increase in delay.

Nadia Mustafa Mohammed Alhag et al. [18] have tried to improve the DES algorithm by extending the key length to 1024 bits, that will be split into 16 keys of 64 bit each, each key is individually generated for the different algorithm sequences. The outcomes of the proposed algorithm were much better than the DES algorithm. This technique has not proposed any ensembled cryptographic algorithm.

Syed Umar et al. [19] have proposed an approach in this paper which deals with the extension of public key and private key encryption using a private key. The private key is generated with the help of ECC and AES algorithm. In this paper, the security of AES algorithm is increased by increasing the key length to 196-bit from 128 bit. Similarly, the number of iterations are increased from 10 to 12. By doing this, the data can be made more secure.

Ako Muhamad Abdullah et al. [20] has proposed none new technique or changes to some existing technique. This paper has summarized the AES algorithm and its security strength. This paper has also shown the techniques several researchers have proposed in the extension of the AES algorithm and its comparison with other algorithms such as DES, Blowfish and 3DES.

Ahmet Zengin et al.[21] have implemented a novel chaos-based encryption algorithm scheme for safe and effective image encryption. To design the solution, the Zhongtang chaotic system has been chosen to be improved because of its strong dynamic features and also dynamical analysis is performed on it.

Using the base of this scheme, a new chaos-based random number generator (RNG) is developed and the applications of the designed RNG in an encryption process are shown over NIST 800-22 randomness tests. S-Box generation algorithm is

created, and the performance tests of S-Box are obtained. Using the designed RNG and S-Box generation algorithms, the new image encryption algorithm based on AES(CS-AES) is developed.

Dheerendra Mishra et al. [22] have tested the security of the proposed authentication scheme of Tu et al. for the Session Initiation Protocol (SIP). Research has explained that an attacker can easily perform the server spoofing, user impersonation and man-in-the-middle attacks on Tu et al.' system. The cryptanalysis of Tu et al.'s scheme thus shows that the security of their scheme is weak. To solve the security vulnerabilities found in Tu et al.'s scheme, the authors proposed a secure and effective authentication scheme for SIP. This research supports mutual authentication and key agreement where a user and a server can accurately identify the legitimacy of each other and can also calculate the session key between them. This scheme satisfies all the required security characteristics, which are showed in the security analysis of the proposed scheme through both informal and formal security analysis. It is concluded that this scheme is more appropriate for practical applications as compared to other schemes.

G. Viswanath et al. [23] have proposed an encryption scheme for securing the cloud environments of big data. The input is divided into blocks of 256-bits. Each 265-bit key block is divided into two blocks of 128-bits. One is plaintext and other is key. The plaintext block is then encrypted by using AES S-box algorithm. 10 rounds are performed on the plaintext by using the substitution and permutation module.

Venkata Koti Reddy Gangireddy et al.[24] have presented a technique for enhancing the security in the cloud, they present a new security model with optimal key selection. They have implemented an improvement in the blowfish algorithm. In this, a k-medoid clustering algorithm is used to cluster the secret message. It is based on the data distance measure. The data is encrypted and stored in the cloud using blowfish encryption technique. To improve the accuracy, the dragonfly algorithm is used.

Mohammed S Mechee et al. [25] have presented this paper to analyzes the security of RAF. The security analysis is divided into two phases. The first phase

examines the output of the entire RAF, including the avalanche text and the correlation coefficient. The second phase examines the quality of the dynamic 3D S-Box generated by the RAF by using the avalanche criterion (AVAL), the strict avalanche criterion (SAC), and the bit independence criterion (BIC). Besides, they also compared the RAF algorithm with the Blowfish algorithm (BA).

Simran bharti Miss Roshni Rathour et al. [26] have presented a security technique for DES algorithm. Before applying the DES algorithm, the substitution layer is added. When the unit of plaintext is replaced with ciphertext, this is called substitution cipher. In this way, two layers are added to the security system, first the attacker has to break the DES algorithm and then he has to break the substation layer. This makes it hard for the attacker to compromise the data.

Wengang Hou et al. [27] have presented an image encryption system which uses AES algorithm. Image is divided into blocks of 128-bits. This technique first mutates the initial block with the help of an initial vector and then applies the AES in cipher block chaining mode to encrypt each block in a sequence. After that, the cipher image and the initial vectors are being sent for decryption via a public information channel. Decryption is done using the secret key and cipher image.Simulation results show that this image encryption system is both safe and high-speed, which can be used as the comparison foundation of newly proposed image encryption systems based on uncontrolled systems.

S Arul Thileeban et al. [28] have presented a new technique using XOR Cipher to encrypt the binary data in images pixel by pixel rather than securing it with an application so that it cannot be exploited or cracked easily.When we encrypt the images pixel by pixel, it is difficult to crack the encrypted data.
The proposed model explains many methods to encrypt the Image using XOR Cipher and the study shows that by using the proposed design, the images are correctly encrypted. The proposed model was tested on various popular images including Mona Lisa, Apollo 11, and NebulaM83, and correct results were produced. These images are well known and standard images to perform these kinds of testing.

Rohit K. Singh et al. [29] have presented a technique which is very light and yet strong enough to secure an extensive amount of data being transferred to the network. This does not require a key that needs to be randomly generated. The plan is to disorganize or replace the characters and make them unreadable by the attacker.The plan is to disorganize or replace the characters and make them unreadable by the attacker. The method changes with an odd or even number of characters. This can be used in many messaging applications where the messages that are to be received are required to be secured. We can let the messages collected in the encrypted format and let the user decrypt it on demand.

Ajay Kushwaha et al.[30] have presented an encryption method named Selective significant data encryption (SSDE) for text encryption. The SSDE provides enough risk to the data encryption process as it selects only significant data out of the entire message. This decreases the encryption time overhead and improves performance. The encryption component is implemented with the help of a symmetric-key algorithm. For this goal, the BLOWFISH algorithm is applied.

Lim Chong Han et al. [31] have presented a secure communication algorithm in this paper which includes three design steps, i.e. the encryption technique, serial-transmission, and encoding technique. The encryption system adopts a combination of Caesar Cipher and XOR encryptions and implemented using the C++ programming language. Afterward, few potential test cases have been tested to verify the strength of the security algorithm, which shows an increase in the security of data transmission in wireless communication without changing the processing time.

From the literature review, we have concluded that there is a significant amount of research done where the researchers have combined symmetric and asymmetric algorithms to make a stronger algorithm based on block size, key size, and many other different parameters. There has been a huge amount of research done where the researchers have combined several cryptographic algorithms together to form a more stronger cryptography algorithm. But, there is an important parameter missing in the research which is plaintext type. Plaintext type can be offline or online.

2.2 Comparative Analysis

In this section, we have discussed the comparative analysis of the existing schemes related to cryptography where researchers have combined several cryptographic algorithms.We have discussed the comparative analysis of the existing schemes related to cryptography where researchers. have proposed single algorithms as well as combinations of multiple algorithms together to secure the data. Table 2.1 explains the comparative analysis of the studied techniques.

TABLE 2.1: Comparative Analysis

Ref	**Methodology**	**Ensembled Technique**	**Machine Learning**	**Plaintext Type**
[3]	Proposed a technique using by combining multiple cryptographic algorithms.This technique comprises of two steps.	Yes	No	No
[4]	Created an ensembled cryptography algorithm for videos,images and text files. Merged AES and DES together to create the cryptography algorithm.	Yes	No	No
[5]	Presented variety of ensembled cryptography algorithms Proposed five encryption algorithms and compared their performances with each other.	Yes	No	No
[6]	Combined AES, DES and MD5 to use their strength together.	Yes	No	No

[7]	Combined three cryptography algorithms to use their strength together Combined RSA, DES, and SHA1 together.	Yes	No	No
[8]	Implemented ensembled cryptography technique using AES and RSA algorithm together to form a more stronger ensembled cryptographic algorithm. The algorithm performance seemed to be good than the plain algorithms.	Yes	No	No
[9]	Proposed an ensembled cryptography scheme developed using JAVA programming language. Used 3DES for encryption of plaintext and RSA for encryption of keys.	Yes	No	No
[10]	Combined the strengths of AES and DES to create a cryptographic algorithm. Used VC++ to develop the system.	Yes	No	No
[11]	Combined ECC and AES to form a new cryptographic algorithm for Wireless Sensor networks.	Yes	No	No

[12]	Created a method to encrypt text by creating a random number generator function. The random number generator function was created manually was the researchers to generate a random number.	No	No	No
[13]	Created a cryptography system for medical systems. This was not an ensembled cryptographic algorithm, that was mainly designed for the security of medical systems. No combination of cryptography algorithm was used in this technique.	No	No	No
[14]	Proposed a new cryptography algorithm by merging two algorithms together to form a more stronger ensembled cryptographic algorithm. The algorithms that have been merged together to form a new algorithm are AES and Blowfish.	Yes	No	No
[15]	Created a new cryptography algorithm by using the combination of RSA and DES algorithm. This algorithm was created for the security of MANETS.	Yes	No	No

[16]	Created a more stronger cryptography algorithm by using the combination of AES and ECC cryptography. AES is used to encrypt the plaintext and ECC is used to encrypt the key.	Yes	No	No
[17]	Created no cryptography algorithm and used 128-bit AES to improve the security of cloud environment.	No	No	No
[18]	Proposed an improvement of DES algorithm by extending its key length to 1024-bits. This algorithm seemed to perform more good.	No	No	No
[19]	Proposed an approach which uses AES and ECC to create a new cryptography algorithm. The private key is generated with the help of AES and ECC algorithm.	Yes	No	No
[23]	Created a new new encryption scheme for securing the cloud environments of big data. The AES S-Box algorithm is used to perform encryption.	Yes	No	No

[20]	Summarizes AES algorithm and compared its security strength with other algorithms such as DES, Blowfish and 3DES.	No	No	No
[21]	Have generated a new, more stronger encryption algorithm for images data. A new random number generation function is created in this system. No paramaters considered while selecting the cryptography algorithms	Yes	No	No

2.3 Identified Research Gaps

After studying the current techniques for single and ensembled cryptography algorithms, we found that most of the researchers have tried to combine multiple symmetric and asymmetric cryptography algorithms together based on block size, key size, and other parameters such as the performance of the algorithm developed.

There is a lack of technique that selects the cryptography algorithm based on the type of text. The type of text can be offline or online and the combination is different in the case of offline or online type of data [1].

2.4 Summary

In this chapter, we studied the different techniques currently published on cryptography where researchers have combined several algorithms together to form a

stronger cryptography algorithm. Our focus was to find out that how the cryptographic algorithms are being merged, what algorithms are being combined, and which parameters are being considered while creating the combinations. We did the comparative analysis of the techniques we studied. We found out that most of the ensembled cryptography algorithms that are being available have considered block size, key size, and key security for a generation. But there is a lack of an important parameter which is plaintext type. The cryptography algorithm combination can be different in the case of a different plaintext type.

In the next chapter, we will discuss our research method and will go through each step we have performed to perform the experiment.

Chapter 3

Research Methodology

In this section, we have discussed the research method of the proposed system. The primary focus of this system is to create a machine learning model, which recommends the combination of cryptographic algorithms based on plaintext type. Type of plaintext can be offline or online. This section has discussed all the steps we have performed to implement the intended machine learning model.

3.1 Experimental Methodology

We have used an experimental method for the technique we proposed. Experimental method mainly comprises an experiment to test the proposed hypothesis. Some hardware and software setup is required to perform the experiment.

3.2 Proposed System Architecture

In this section, we are going to explain the research method in detail which we have performed to recommend the combination of cryptographic algorithms. For this purpose, we have used experimental methodology. In order to recommend the combination of machine learning models, we have used four machine learning

classifiers.

The set of steps we have performed can be seen in figure 3.1.

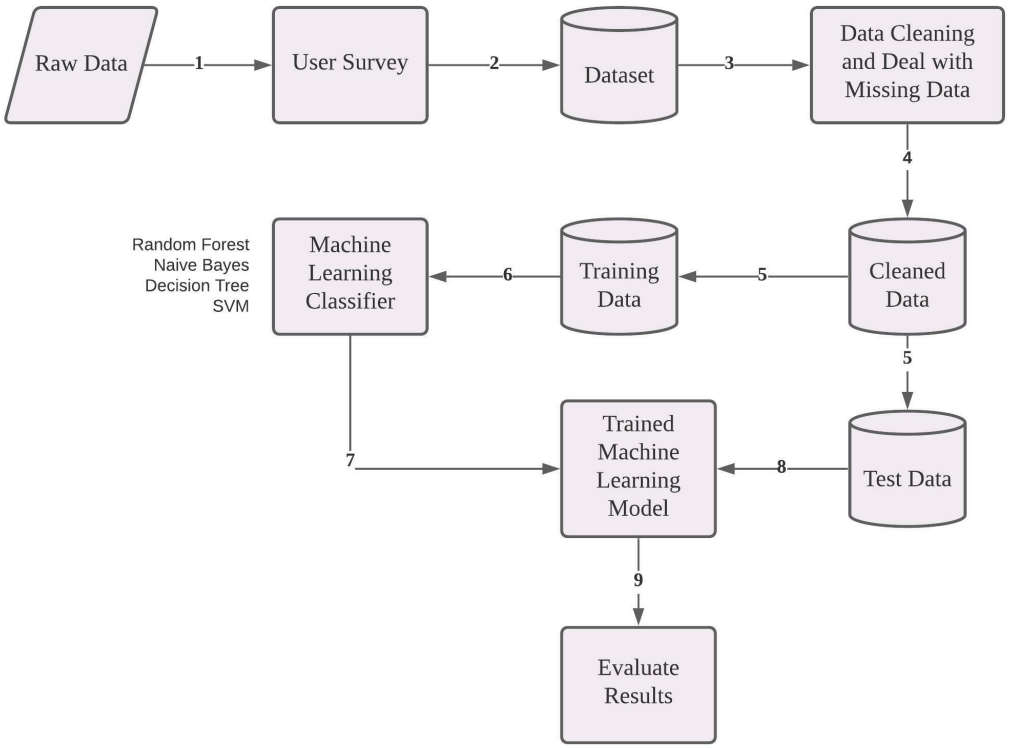


FIGURE 3.1: System Architecture Diagram

3.2.1 Dataset Creation Process

As mentioned earlier, there was no benchmark dataset available for this experiment according to the literature review we did, which recommends the combination of cryptographic algorithms for a sample of plaintext based on its type. So we collected 5000 plaintext sentences from different online and offline sources and converted them into feature vectors. The offline and online sources are mentioned in the table later on. Subject experts validated the dataset through a survey. We distributed the data among 5 different subject experts from the domain of information security. Four of them were penetration testers in separate organizations. One of them was a malware researcher at a notable organization.

We distributed the data among subject experts and asked them about the most suitable combination of algorithms they think can be applied for each sample

of plaintext. Subject experts had to provide their assumption about the most suitable single and combination of algorithms they think for all the samples of plaintext in the dataset. Figure 3.2 explains the process of the user survey that was conducted to create the dataset.

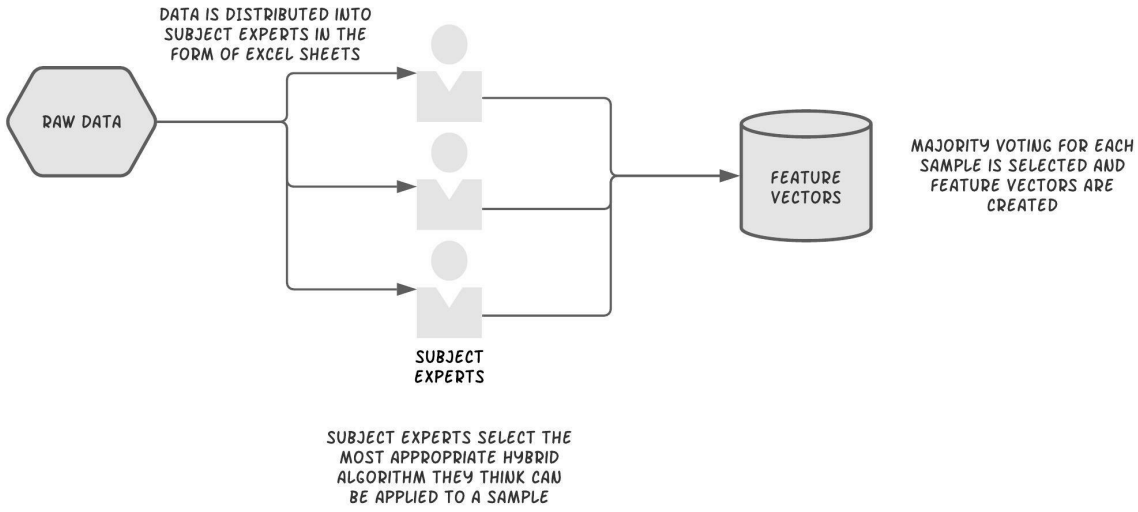


FIGURE 3.2: User Survey

We distributed data in an excel format to the subject experts and were having the features mentioned in table 3.2.1

TABLE 3.1: Sample Data for Survey

Feature name	**Reason**
data_text	plaintext to be encrypted
data_nature	type of the plaintext (offline/online).
data_source	The source from where plaintext was collected from.
single_algorithm	single algorithm choice selected by subject expert.
hybrid_algorithms	hybrid algorithm choice selected by subject expert.
text_size	number of characters in the data text.

After taking recommendations from each of the subject experts, the final dataset was created by taking the most recommended combination of algorithms for each plaintext sample.After taking recommendations from each of the subject experts, the final dataset was created. The most voted value was adjusted for each sample

of the dataset. Such as if one sample had 3 out of 5 votes, then that value was adjusted into the final sample. Our dataset comprised 5000 samples. We converted the whole dataset into numerical notation for the ease of the machine learning process. The notations we used are described in tables below.

TABLE 3.2: Type of Plaintext

plaintext type	**numerical notation**
offline	0
online	1

The hybrid combinations we used in our data sets were also collected from the subject experts mentioned in the section and from the literature review. The subject experts were being asked about the most used cryptography algorithms in the industry and their recommendations were collected on the most suitable combinations of those algorithms. The combinations we used and their numerical notations are mentioned in table 3.2.1

TABLE 3.3: Hybrid Cryptography Algorithms and Their Numerical Notations

numerical notation	**hybrid cryptography algorithm**
1	RC4
2	DES+3DES+BlowFish
3	AES+3DES
4	AES+DES
5	AES+Blowfish

The data collection was a heavy process and it took a lot of time to collect the data. We collected the sample data from several online and offline sources. Offline sources mean where we are not connected to the internet and online source mean where we are connected to the internet. The data which is traveling over a network is called online data whereas the data which is at rest or not traveling over a network is called offline data.The data which is traveling over a network is

called online data whereas the data which is at rest or not traveling over a network is called offline data. We also used a numerical notation for the sources from where we collected the data from. The sources and their numerical notations are mentioned in table 3.2.1.There were four major sources from where we collected the data. Cloud database, Desktop applications, Offline database, and web applications. Cloud databases and web applications were online data sources, whereas desktop applications and offline databases are offline data sources.

TABLE 3.4: Source of Data

plaintext type	**numerical notation**
1	Cloud Database
2	Desktop Application
3	Offline Database
4	Web Application

3.2.2 Data Preprocessing

This Step involves cleaning the data to fit into a machine learning algorithm for better accuracy and performance. We have cleaned the data and performed some preprocessing operations on the data so that the machine learning algorithm can easily train a model without any or fewer errors. Data preprocessing is a continuous cycle having four steps which can be seen in figure 3.3.

Data cleaning is a process where we remove unnecessary data from the dataset, which may have a negative impact on the machine learning model prediction. **Data integration** is a process where we combine data from multiple sources together to form one dataset. Such as we combined plaintext sentences from several sources together. **Data Transformation** is converting the data into one form from another such as converting the English labels into a numerical format for the ease of machine learning model. **Data reduction** is removing the unnecessary features from the dataset which do not have any impact on the machine learning model prediction. The steps mentioned in the figure 3.3 are repeated until the data is in shape to be trained by the machine learning model.

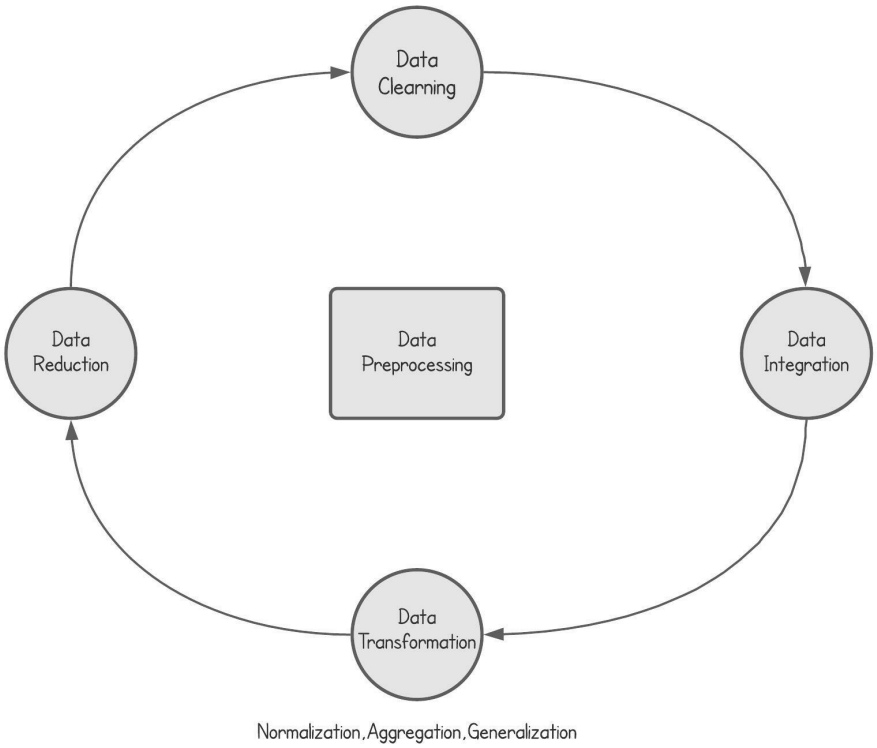


FIGURE 3.3: Data Preprocessing

3.2.3 Model Training

This is one of the crucial steps of our system and involves training the machine learning model by using the training data. Before this step, we will divide the cleaned data into a training dataset and a test dataset. The percentage we used for training data is 80% and for test data is 20%. After division, we will fit the training dataset into the machine learning algorithm for training the classifiers. The machine learning algorithm will train itself on the provided data and will generate the trained model that will be used in further steps to perform the evaluation. Machine learning classifiers that we have used are discussed in section 3.4. We have used Support Vector Machines, Random Forest, Decision Tree, and Naive Bayes classifiers for training our model as these are the most used classifiers in research. The model training process can be seen in figure 3.4

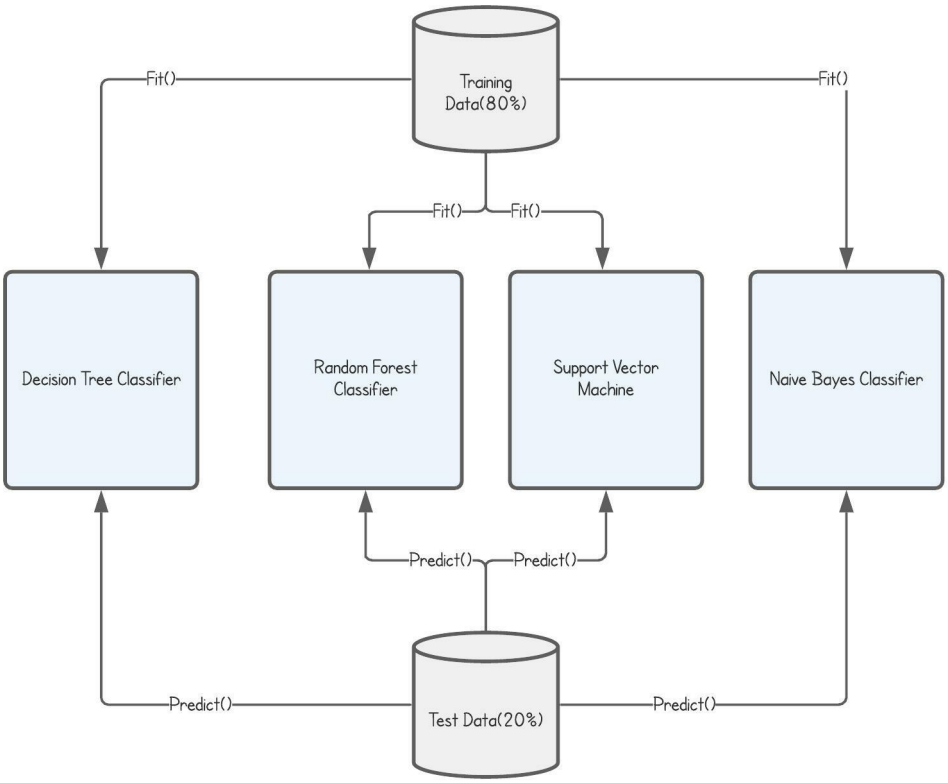


FIGURE 3.4: Model Training

3.2.4 Evaluation of Trained Model

This is the last and most important step of our experiment, which involves testing the trained model using the test dataset. The percentage of test data we have used is 20% data. We have trained four machine learning classifiers to test and compare their results among each other.Support Vector Machine, Random Forest, Decision Tree, and Naive Bayes classifier have been trained to evaluate their result.

We will test the trained model by providing the test dataset to it. Later on, we will evaluate the results using the metrics we have selected for evaluation. The metrics which we have selected for evaluation are discussed in section 4.1. If the results are satisfactory and we are getting a good accuracy score, then we will complete our results. Otherwise, we will repeat all the steps to achieve a good accuracy score.

3.3 Motivation of using machine learning

Machine learning classifiers are used in a variety of applications. Researchers are using machine learning to benefit almost every field. Machine learning is also playing an important part in the domain of cryptography and information security as well [32]. Researchers have tried to apply machine learning in for different techniques, such as prevention of attacks on data and detecting the attacks on data.such as prevention of attacks. We did a literature review on ensembled cryptography and tried to find out whether there are any techniques available which used plaintext type as a parameter for ensembled cryptography.
After doing the literature review, we identified that there is a lack of techniques available which uses machine learning to recommend the combination of cryptographic algorithms. This gave us the motivation to use machine learning to create a machine learning model, which can recommend a hybrid cryptography algorithm based on the type of plaintext.

3.4 Machine Learning Classifiers used for Training

The selection of the correct classifier for the training phase is the most vital phase in our work as the classifiers will decide the accuracy of the dataset we created according to user study and also the accuracy of the machine learning model we want to create.

We have used supervised machine learning to train the models. We did a thorough literature review in the field of network security and cryptography to find out which of the supervised machine learning algorithms are widely used by researchers to make the data secure. We came across many algorithms and selected these three algorithms out of those algorithms as these were the most used algorithm by almost every researcher and gave good results.

1. Decision Trees.

2. Random Forest
3. Support Vector Machine
4. Naïve Bayes

3.4.1 Decision Trees

The Decision Tree algorithm applies to the class of supervised machine learning algorithms. The decision tree algorithm is widely used in the field of network security and cryptography for solving regression and classification problems.

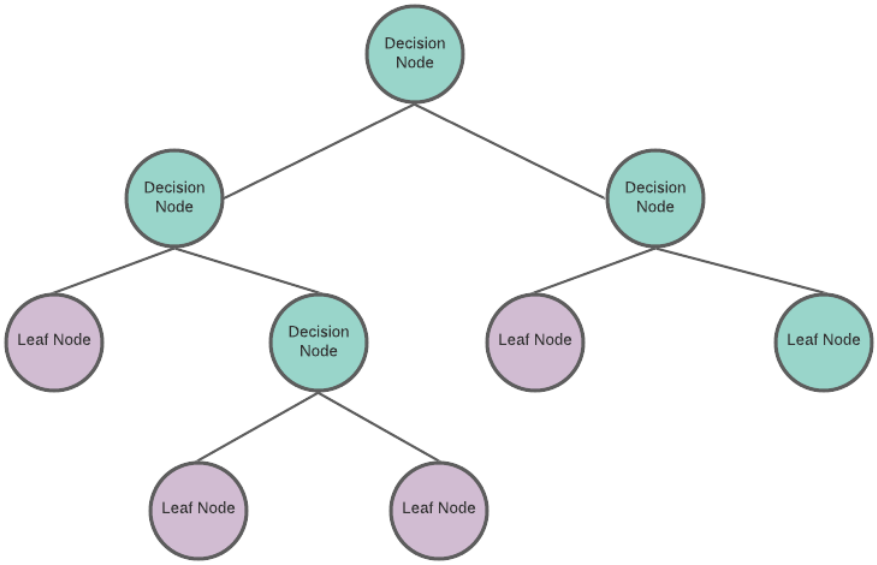


FIGURE 3.5: Decision Tree

The idea of the Decision tree algorithm is to build a model that can be used to predict the class of the target variable by learning simple decision rules gathered from past data. It starts from the root of the tree and moves down to the leaf nodes till there are no leaves left. [33].In Decision Trees, to predict a class label for a record, we begin from the root of the tree. We match the contents of the root attribute with the record's attribute. Using the results of the comparison, we move towards the branch corresponding to that value and jump to the next node. Decision trees are considered the most powerful classification algorithms for classification and prediction problems. To train these, we need to take care of the

hyper parameters we are providing to them. Hyper parameters play an important role in the performance of Machine learning classifiers.

3.4.2 Random Forest

Random Forest (RF) is one of the machine learning algorithms which are widely used to solve regression and classification problems. Random forest uses an ensemble learning technique that combines many machine learning classifiers to solve complex problems in the machine learning domain.
This algorithm comprises more than one decision tree. This algorithm uses one of the two methods to train the algorithms i-e bagging or bootstrapping. Bagging is an ensemble algorithm that tends to improve the accuracy of machine learning algorithms.

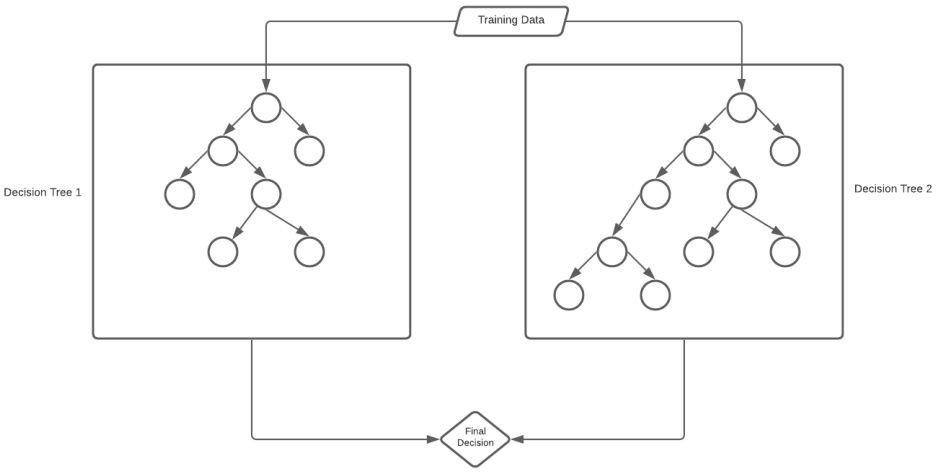


FIGURE 3.6: Random Forest Algorithm

Random Forest algorithm makes its final decision based on the output from all the decision trees it creates. It takes the mean or average of the outputs from all the trees [34]. The more trees we have, the more increased precision there will be. This algorithm eliminates the shortcomings of the decision tree algorithms and reduces the chances of overfitting of datasets and gives an increased precision.

3.4.3 Support Vector Machine

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms among the most used classifiers for Supervised Machine Learning, which is used for Classification and Regression problems [35]. But mostly, it is used for Classification problems in Machine Learning. It can be used to solve binary or multi classification problems

The main aim of the Support Vector Machine (SVM) algorithm is to create the best line or decision boundary that can separate n-dimensional space into classes so that we can easily put the new data point in the correct class in the future. So that the data point can be easily placed into correct class in the future. This most suitable decision boundary is called a hyperplane.

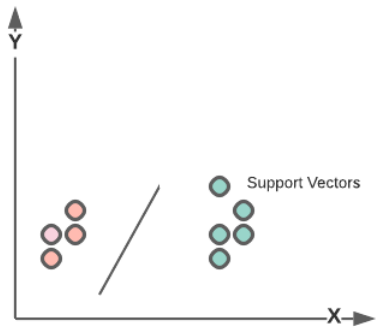


FIGURE 3.7: Support Vector Machine

SVM takes the extreme vectors that help in creating the hyperplane. These cases are termed as support vectors, and henceforth algorithm is called a Support Vector Machine.

3.4.4 Naïve Bayes

Naive Bayes algorithm [36] is a simple algorithm that is used to perform classification on the data. It belongs to the statistical family of classification algorithms and falls under supervised machine learning algorithms.This algorithm is widely used to predict and test classification type problems. This algorithm has seemed to perform well in the case of multiclass problems as well.

Naive Bayes uses the Bayes theorem to calculate the probability of a specific case from a given set of feature vectors.

$$P(c|r) = P(r|c)P(c)/P(r) \tag{3.1}$$

- P(c r) is the posterior probability of target class given attribute.
- P(c) is the prior probability of class
- P(r c) is the likelihood which is the probability of attribute given class.
- P(r) is the prior probability of attribute.

3.5 Experimental Setup

Our proposed technique requires building a machine learning model and building a machine learning model requires good computational resources. A good hardware setup is required to carry out the experiment. Table 3.5 has the hardware and software configuration we have used to carry out the experiment. We have implemented our model in machine learning library and have used the well known IDE for the development of machine learning model. The library we used is sci-kit learn.

Table 3.5: Hardware and Software Configuration

Hardware Configuration	
CPU	Intel(R) Core(TM) i7-5600U CPU @ 2.60GHz 2.59 GHz
Operating System	Windows 10
RAM	16GB
System Type	64-bit
Software Configuration	
Machine Learning Library	Sci kit Learn
Programming Language	Python
IDE	Jupyter Notebook

3.6 Hyperparameters Tuning for Classifiers

When we work with machine learning models, we have many design choices to define the model of your machine learning algorithm. This often occurs that we don't immediately do the optimal configuration for our machine learning model and have to explore the range of possibilities to get excellent results.

The parameters which make the model architecture are known as hyper parameters. In the sci-kit learn library, there are always pre-defined or default values for all the hyper parameters. We have changed some parameters to optimize our results. The important set of configured parameters in order to carry out the experiment are mentioned in table 3.6

The hyper parameters play an important role in the performance of machine learning algorithm. When we fine tune these algorithms, they impact the model training process by increasing or decreasing the performance. The hyper parameters we mentioned in table 3.6 are the most important hyper parameters for these algorithms.

3.6.1 Hyperparameters for Decision Tree

Table 3.6 shows the settings of the hyperparameters of the decision tree classifier. **Gini criterion** is the measure of impurity. It is also called Gini index. It calculates a probability of a feature that it will be put into a wrong class when selected randomly from the dataset.

TABLE 3.6: Decision Tree Hyper parameters Configuration

Parameter Name	**Configuration**
criterion	gini
max_depth	150
min_sample_split	2
min_sample_leaf	1
max_features	None

For example: if we have 5 samples of RC4 in our dataset and 5 samples of DES+3DES+BlowFish the level of impurity is 0.5. and if we have all the 10 sample belonging to RC4, then there is no chance that RC4 is incorrectly classifier and level of impurity is 0.

The maximum depth of the tree is controlled by **max_depth**. Lets say if we have the setting of max_depth to 1 then the decision tree will not expand more than level 1.

min_samples_split means that how many samples should be inside a node to expand it further. Let's say we have set this parameter to 10. This means that a node should have 10 samples to expand them further into child nodes.

min_samples_leaf means a node should have a specific number of samples to become a leaf node. Let's say that we have 7 samples and min_samples_leaf is set to 2. Then the expand will not be allowed, because one of the leaf node should have less than 2 samples, which is not allowed.

max_features determines the number of features that we select to get the best split. If this parameter is not set, then the algorithm will consider all the features to make the best split.

3.6.2 Hyperparameters for Random Forest

Table 3.7 shows the settings of the hyper parameters of random forest classifier.

Table 3.7: Random Forest Hyperparameters Configuration

Parameter Name	Configuration
n_estimators	1000
max_features	auto

When we build a random forest classifier, multiple decision trees are made and at the end, the best vote from these is selected as the conclusive answer. **n_estimator** determines the number of decision trees we want to make to take the best vote

from.

max_features determines the number of features that we select to get the best split. If this parameter is not set. This parameter purpose is same for decision tree and random forest classifier.

3.6.3 Hyperparameters for SVM

Table 3.8 shows the settings of the hyperparameters of the support vector machine classifier. We have tuned the values of the kernel, C, and gamma variables and have achieved a fine accuracy score.The most important hyperparameters, when it comes to the performance of SVM are kernel type and the gamma. C is also an important parameter when we want to achieve a high accuracy score. The value of kernel is Radial, C is set to be 1.0 and gamma is scale

TABLE 3.8: SVM Hyperparameters Configuration

Parameter Name	**Configuration**
kernel	Radial
C	1.0
gamma	scale

Radial kernel is the default kernel set by scikit learn library and it is defined by the below formula.

$$K(x, x') = ||x - x'|| \tag{3.2}$$

| | x-x' | | is the length of a line segment between two feature vectors. This is also called Squared Euclidean distance.

Gamma determines the value or influence of a single training sample.If we increase the value of the gamma, The points should be closer to each other to affect the model. its value can be set manually and its default value is defined by the formula

$$\gamma = \frac{1}{nfeatures * \sigma^2} \tag{3.3}$$

C basically determines the margin of the hyperplane. This tells that how much we want to avoid misclassification of the training data.

3.6.4 Hyperparameters for Naive Bayes

Naive bayes algorithm does not have many parameters to fine-tune. We have used Gaussian Naive Bayes for the implementation where it is considered that each class is normally distributed [37].

3.7 Summary

In this chapter, we discussed the research method which we have followed to perform the experiment. We have also seen the hardware and software configuration for the experiment. We trained four different machine learning classifiers on the dataset we created and evaluate their results. The machine learning classifiers we have trained are Random forest, Decision tree, SVM, and Naive Bayes. These algorithms were picked after doing the literature review on the most popular supervised machine learning algorithms being used by the research community [38]. In the next chapter, we are going to evaluate our results on different metrics and discuss the results.

Chapter 4

Results and Discussions

In this chapter, we are going to evaluate our proposed system. This system intends to recommend a combination of cryptographic algorithms based on plaintext type using supervised machine learning techniques. We have used different supervised machine learning classifiers to achieve this and also intend to compare their performances with each other to select the best one out of them. There was no gold standard dataset available for this hypothesis as we were targeting the plaintext type and combination of different algorithms in the dataset as well, so we created the dataset by user survey. We have discussed the dataset creation process in section 3.2.1. In this chapter, we will talk about the evaluation parameters we have used for different classification algorithms and will evaluate and discuss the results we got from our classifiers.

4.1 Model Evaluation Measurements

It is normally difficult to decide which evaluation parameter to choose for a problem. Each of the parameters has distinct features that measure various aspects of the classifier being evaluated. Mostly, the performance evaluation of a machine learning classifier is evaluated by the predicted accuracy of the model, which is not alone enough to declare the model as a good machine learning model. Mostly the

researchers use the accuracy of a machine learning model to declare it as a performant model or not. But there are other many certain parameters as well which are involved in the performance measurement of the machine learning model. Machine learning performance evaluation involves certain trade-offs between true positive rate and false-positive rate. Precision, recall, and F-measure is commonly used as evaluation parameters of machine learning model [39]. Below are the performance evaluation metrics we have used to evaluate our classifiers. We have a multi-class problem so we only choose those evaluation metrics which are relevant to our problem [38].

4.1.1 Confusion Matrix

Confusion Matrix is one of the basic evaluation measure of machine learning classifiers. Confusion matrix is a technique which summarizes the performance of a machine learning classification algorithm. When we calculate a confusion matrix, we can see that how our algorithm is performing and what errors it is producing. It basically shows the number of correct and incorrect predictions by a machine learning algorithm. It is a figure formed by visualization of correct and incorrect classified classes. For our case, there were 5 classes which contained different combinations of cryptographic algorithms for encryption of text data.Confusion Matrix has 4 classes.

True Positive (TP)

True positive means that the model has classified a sample into the correct class. For example, Positive was predicted as positive.

False Negative (FN)

False-positive means that the model has classified a sample into the incorrect class. For example, Negative was predicted as positive.

True Negative (TN)

False positive means that the model has classified a sample into the incorrect class. For example, Positive was predicted as negative.

False Positive (FP)

True positive means that the model has classified a sample into the correct class. For example, Negative was predicted as Negative.

4.1.2 Accuracy

Accuracy is the ratio of the correct predictions to the total number of predictions made by the machine learning classifier. This performance measure helps to drive more meaning out of your machine learning classifier. It can be calculated by using equation 4.1.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \ , \tag{4.1}$$

4.1.3 Precision

Precision is the ration of total correct predictions made by the classifier to the sum of total correct predictions made by the classifier and total false positive predictions. It can be calculated by using equation 4.2

$$Precision = \frac{TP}{TP + FP} \ , \tag{4.2}$$

4.1.4 Recall

Recall is the ratio of the total number of classes that were classified correctly to the total number of classes that were classified correctly or incorrectly. It can be calculated by using equation 4.3.

$$Recall = \frac{TP}{TP + FN} \ , \tag{4.3}$$

4.1.5 F1 Score

F-measure or F1 score is the harmonic mean of precision and recall. The value is near to 0.0 for the worst F1 score and near to 1.0 for the perfect F1 score [39]. It can be calculated by the equation 4.4.

$$Recall = 2 * \frac{Precision * Recall}{Precision + Recall} , \tag{4.4}$$

4.1.6 Model Training and Test Time

The time which a machine learning classifier takes to train and test itself is also important. We will also consider the model training and test time while evaluating our machine learning classifiers. If a machine learning classifier is taking more time to train and test the data we provide to it. That means in the future when we increase the data size, the performance will be affected. Our goal is to create a machine learning model which recommends the combination of cryptographic algorithm without little impact on the training and test time.

4.2 Classification

In our work, we have used four well-known machine learning classifiers to recommend the combination of cryptographic algorithm based on plaintext type. The type of plaintext is also used as an attribute in the dataset. The classifiers we have used are Decision Tree [32], Random Forest [35], Support Vector machine [33] and Naive Bayes [40].

Our classification problem is a multi class problem which has 5 classes and those classes contain different combinations of cryptography algorithms.

Hyper parameters are one of the most important parameters in the machine learning process. Hyper parameters control the machine learning process and can have a major impact on the model performance. We have used several hyper parameters

in our classification algorithms as well. As we have used sci kit learn library, that library sets the values of the hyper parameters by default. If we are not getting excellent results, we can change the values of those hyper parameters as well. We have discussed the use of those parameters in section 3.6 in detail.

We divided the dataset into 20% test data and 80% training data. As the dataset was created by user survey, so currently we have a few data to train our machine learning classifiers. We used sci-kit learn library to train and test our machine learning classifiers. Later on, we perform the evaluation using different performance metrics used to measure the performance of a machine learning model. We only picked those parameters which are valid according to multi class classification problem [38].

4.3 Classification Results and Evaluation

This is the most important section of our whole research, where we compare the results of the classifiers we have trained and then compare those results with each other based on performance metrics we selected to evaluate. In this section, we give the conclusion about which of the classifiers performs best for our given dataset.

First, we compare the accuracy of all the classifiers with each other. Accuracy is considered the most popular way of evaluating the machine learning model and it is directly calculated by the confusion matrix [38], which will be discussed later as well.The comparison of accuracy of each individual class is shown in figure 4.1.

Figure 4.1 shows the comparison of each individual class's accuracy with each other. It can be seen that Decision tree, Random Forest, SVM, and Naive Bayes classifiers have the accuracies of 92.04,98.01,99.6, and 84.29. Naive Bayes has the lowest accuracy among all the classifiers for class 0. SVM has performed best for class 0 with the highest accuracy of 99.60.

For the Class 1: Decision tree, Random Forest, SVM, and Naive Bayes classifiers have accuracies of 80.80,60.80,54.40, and 80. SVM has the lowest accuracy among all the classifiers for class 1. Decision Tree has performed best for class 1 with the

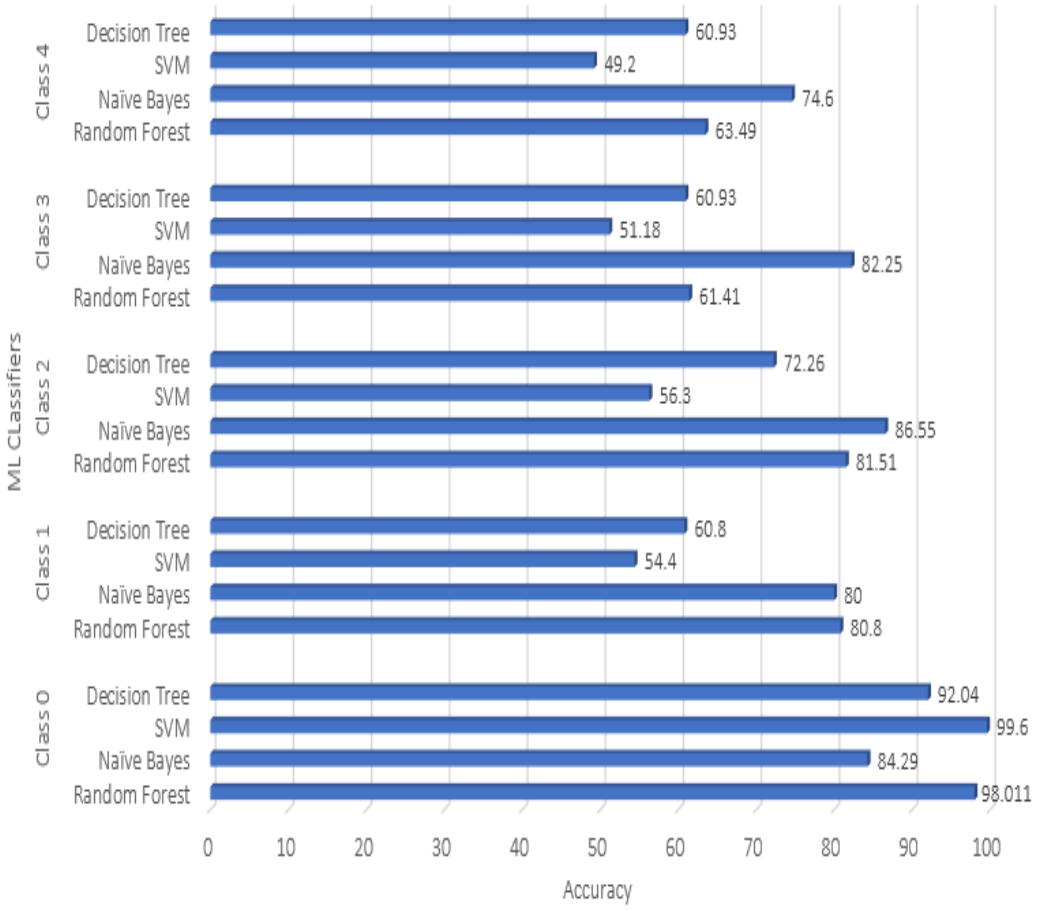


FIGURE 4.1: Accuracy Comparison of Classes

highest accuracy of 80.80.

For Class 2: Decision tree, Random Forest, SVM, and Naive Bayes classifiers have accuracies of 72.26,81.51,56.3, and 86.55. SVM has the lowest accuracy among all the classifiers for class 2 which is 56.3. Naive Bayes has performed best for class 2 with the highest accuracy of 72.26.

For Class 3: Decision tree, Random Forest, SVM, and Naive Bayes classifiers have accuracies of 60.93,61.41,51.18, and 82.25. SVM has the lowest accuracy among all the classifiers for class 3 which is 51.18. Naive Bayes has performed best for class 3 with the highest accuracy of 60.93.

For Class 4: Decision tree, Random Forest, SVM, and Naive Bayes classifiers have accuracies of 60.93,63.49,49.2, and 74.6. SVM has the lowest accuracy among all the classifiers for class 4 which is 49.2. Naive Bayes has performed best for class 4 with the highest accuracy of 74.6

Now we compare the overall accuracy score for the classifiers, which tells us that

which algorithm has performed best in terms of accuracy. Figure 4.2 tells us the overall accuracy score of each algorithm.

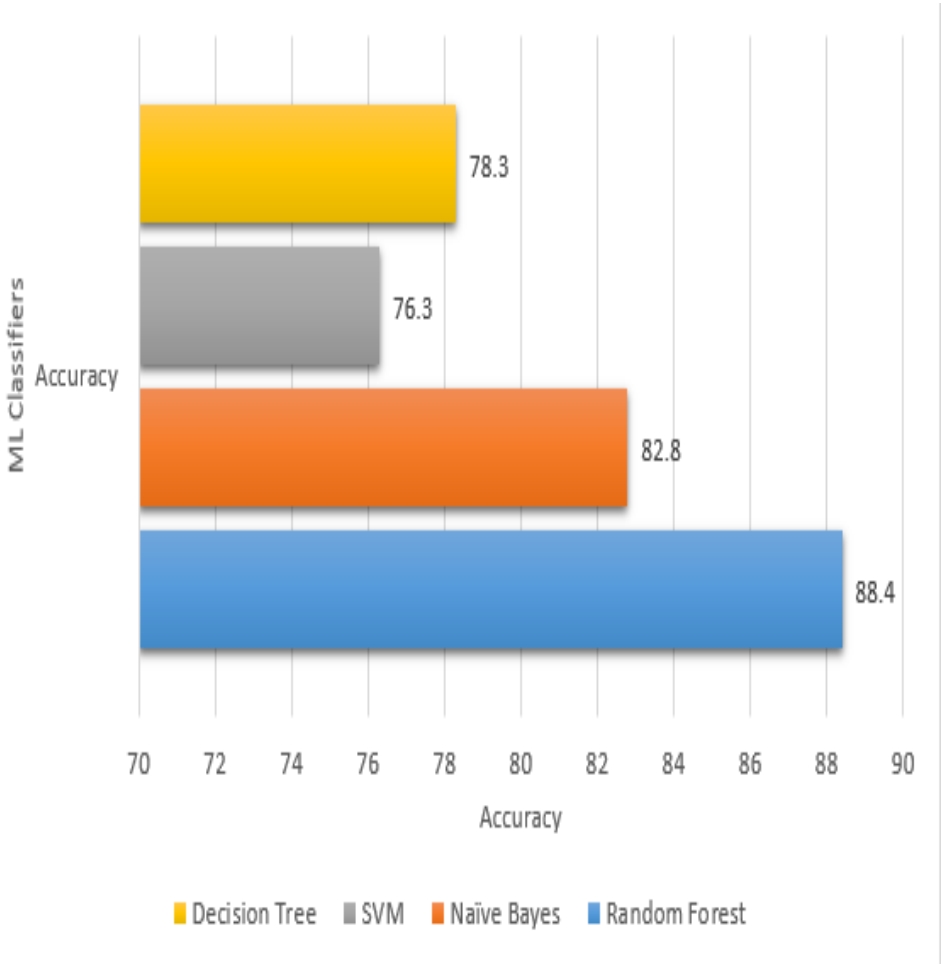


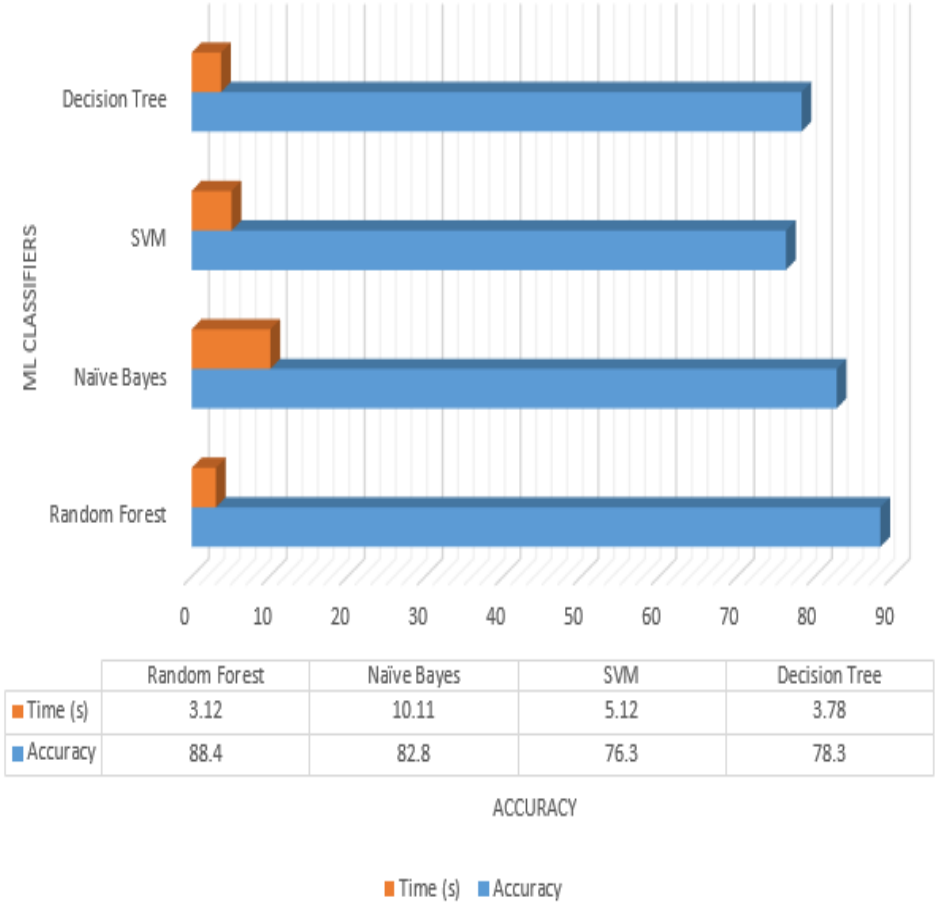
FIGURE 4.2: Overall Accuracy Comparison of Classifiers

Time is an important factor when it comes to machine learning classifiers. When a machine learning classifier is performing very well in terms of accuracy, but it is taking a lot of time to train and test itself on a very small size of data, then the model performance will be impacted a lot when the data size will be increased. we have also considered the model training and test time in this so we can check that if the model is giving good accuracy, whether it is achieving that accuracy in a good amount of time or not. The random forest algorithm have given the best accuracy among all the classifiers but it is also important to check whether the classifier has trained and tested itself in a good amount of time or not.

Figure 4.2 tells us about the overall accuracy score each classifier has achieved. We can see that the Random Forest algorithm has performed the best of all by

achieving 88.4% accuracy. The decision tree classifier has achieved 78.3% accuracy. Support vector machine has achieved 76.3% Accuracy and Naive Bayes algorithm has achieved 82.8% Accuracy. The Support Vector Machine has shown the lowest accuracy on the dataset. The Random Forest algorithm gave the best performance among all the algorithms and achieved a good accuracy score near to 90.

We have also done the comparison of all the algorithms w.r.t accuracy and time. Figure 3 shows that how much time has a classifier taken to train itself on the provided training data and then test the data and give the results.



	Random Forest	Naïve Bayes	SVM	Decision Tree
Time (s)	3.12	10.11	5.12	3.78
Accuracy	88.4	82.8	76.3	78.3

FIGURE 4.3: Comparison of Time and Accuracy

We discussed earlier that the Random Forest algorithm has achieved the best accuracy among all the other classifiers. However, if we compare the time taken by Random Forest to train and test the provided data. It has taken the most time to complete the machine learning process. Whereas if we see the other algorithms, they have achieved very good accuracy with very little time. If we compare the accuracy w.r.t time, Decision Tree has performed the best of all the algorithms by

achieving 91.3% accuracy in just 3.78 seconds. But as we said earlier, the main focus of our system is to focus on the quality of encryption, so we still consider the Random Forest algorithm the best performant of all despite the time it has taken as it has the best accuracy among all of them.

Previously, we evaluated the Machine learning classifiers by accuracy. After comparison of the accuracy of the classifiers. It was concluded that the Random Forest algorithm performed best in terms of accuracy. However, if we consider the time performance with accuracy, the Decision Tree algorithm stood best among all of them.

Now we are going to compare the performances of all of these classifiers using a confusion matrix. The confusion matrix is an excellent way to evaluate the performance of a machine learning classifier as it clearly shows the number of correct and incorrect predictions made by the classifier. figures 4.4, 4.5, 4.6 and 4.7 show the confusion matrix of all the classifiers in detail.

		SVM Accuracy				
Actual	Class 0	501	1	1	0	0
	Class 1	56	68	1	0	0
	Class 2	49	2	67	1	0
	Class 3	61	0	1	65	0
	Class 4	64	0	0	0	62
		Class 0	Class 1	Class 2	Class 3	Class 4
		Predicted				

FIGURE 4.4: Confusion Matrix for SVM Classifier

Figure 4.4 shows the confusion matrix for the SVM algorithm. Figure 4.4 tells us that for Class 0, 501 out of 503 predictions were correct, and only 3 predictions were falsely predicted by the classifier. Similarly for Class 1, 68 out of 125 predictions were correct, and only 57 predictions were falsely predicted by the classifier. For class 2, 67 predictions out of 119 were correctly predicted by the classifier and only 52 predictions were not correctly predicted by the classifier. For class 3, 65 predictions out of 127 were correct and 62 were incorrect. For class 4 there were 62 correct predictions out of 126 predictions and 64 incorrect predictions. So SVM seemed to perform well according to the confusion matrix of the algorithm.

		Decision Tree				
Actual	Class 0	463	12	8	10	10
	Class 1	41	76	6	0	2
	Class 2	29	4	86	0	0
	Class 3	44	3	0	78	2
	Class 4	44	0	0	2	80
		Class 0	Class 1	Class 2	Class 3	Class 4
		Predicted				

FIGURE 4.5: Confusion Matrix for Decision Tree

Figure 4.5 shows the confusion matrix for the Decision Tree algorithm. We can see that for Class 0, 463 out of 503 predictions were correct, and only 40 predictions were falsely predicted by the classifier. Similarly for Class 1, 76 out of 125 predictions were correct, and only 49 predictions were falsely predicted by the classifier. For class 2, 86 predictions out of 119 were correctly predicted by the classifier and only 33 predictions were not correctly predicted by the classifier. For class 3, 78 predictions out of 127 were correct and 49 were incorrect. For class 4 there were 80 correct predictions and 46 incorrect predictions made by the classifier. Decision Tree algorithm also performed very well in terms of confusion matrix. For class 0 it made more correct predictions, while for class 3 the number of correct predictions were reduced compared to SVM.

		Random Forest				
Actual	Class 0	493	6	0	0	4
	Class 1	22	101	2	0	0
	Class 2	20	2	97	0	0
	Class 3	28	0	0	99	0
	Class 4	32	0	0	0	94
		Class 0	Class 1	Class 2	Class 3	Class 4
		Predicted				

FIGURE 4.6: Confusion Matrix for Random Forest

Figure 4.6 shows the confusion matrix for the Random Forest algorithm. We can see that for Class 0, 493 out of 503 predictions were correct, and only 10 predictions were falsely predicted by the classifier. Similarly for Class 1, 101 out

of 125 predictions were correct, and only 24 predictions were falsely predicted by the classifier. For class 2, 97 predictions out of 119 were correctly predicted by the classifier and only 22 predictions were not correctly predicted by the classifier. For class 3, 99 predictions out of 127 were correct and 28 were incorrect. For class 4 there were 94 correct predictions and 32 incorrect predictions made by the classifier.

The correct number of predictions made by the Random Forest algorithm was greater than the algorithms that are being discussed before. random forest algorithm performed well and had an equal number of correct and incorrect predictions for each class.

		Naïve Bytes				
Actual	Class 0	424	15	13	30	21
	Class 1	10	100	6	6	3
	Class 2	12	0	103	4	0
	Class 3	10	2	8	107	0
	Class 4	22	2	4	4	94
		Class 0	Class 1	Class 2	Class 3	Class 4
		Predicted				

FIGURE 4.7: Confusion Matrix for Naive Bayes

Figure 4.7 shows the confusion matrix for the Naïve Bayes algorithm. We can see that for Class 0, 424 out of 503 predictions were correct, and 49 predictions were falsely predicted by the classifier which is very low as compared to the performance of the previous classifiers. For Class 1, 100 predictions were made correctly by the classifier where as 25 predictions were incorrect. For class 2, 103 predictions out of 250 were correctly predicted by the classifier and 16 predictions were not correctly predicted by the classifier. For class 3, 107 predictions out of 241 were correct and 20 were incorrect. For class 4 there were 94 correct predictions made by the classifier and 32 incorrect predictions.

In the Naive Bayes classifier, correct predictions were reduced as compared to the other three classifiers for Class 0. Whereas increased a little for other classes.
Now we compare the number of correct predictions and the number of incorrect

predictions made by each classifier for each class in a graphical format to get a better picture. Figure 4.8 shows the number of correct predictions of each classifier class-wise and Figure 4.9 shows the number of incorrect predictions made by each classifier separately for each class.

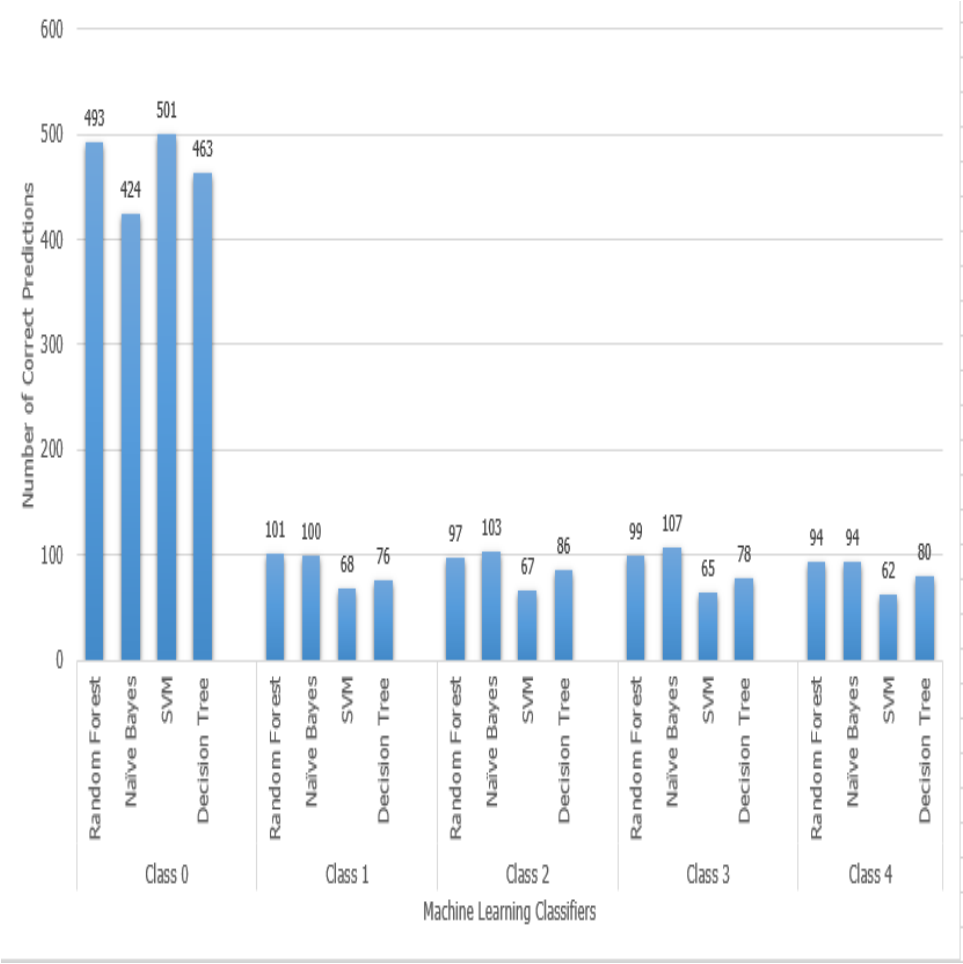


FIGURE 4.8: Correct Predictions of Classifiers Class Wise

We can see from figure 4.9 that the number of correct predictions made by the random forest classifier is higher than the other classifiers. For class 0, the random forest has made 490 correct predictions. Similarly, for class 1, the random forest has made 101 correct predictions. For class 2 the algorithm has made 97 correct predictions. For class 3, 99 correct predictions are being made by the random forest algorithm and for class 4 the algorithm has made 94 correct predictions. Now let us see the number of incorrect predictions made by each classifier.

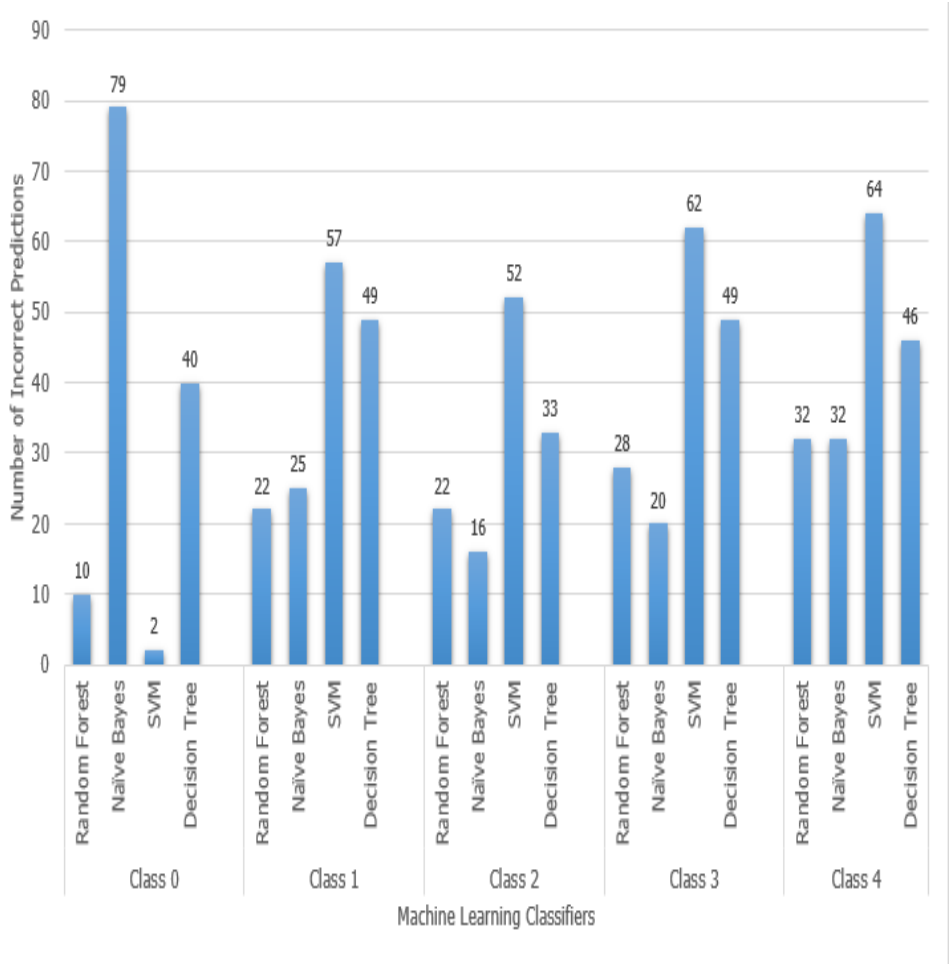


FIGURE 4.9: Incorrect Predictions of Classifiers Class Wise

When we talk about the number of incorrect predictions, we can see from figure 4.9 that Random Forest has given the lowest number of incorrect predictions out of all the four classifiers. For class 0, Random forest has given 10 incorrect predictions. For class 1, the random forest has given 22 incorrect predictions. For class 2, the classifier has given 22 incorrect predictions. For class 3 there are 28 incorrect predictions and for class 4 there are 32 incorrect predictions.

Figure 4.10 finalizes the performance of the Confusion matrix in the form of a graph by showing the total number of correct predictions made by each classifier and the total number of incorrect predictions made by each classifier.

We can see from figure 4.10 that the number of correct predictions made by the Random forest algorithm is higher than any of the other algorithms. Similarly, the number of incorrect predictions made by the random forest algorithm is less than all of the other algorithms. So we conclude that by the confusion matrix

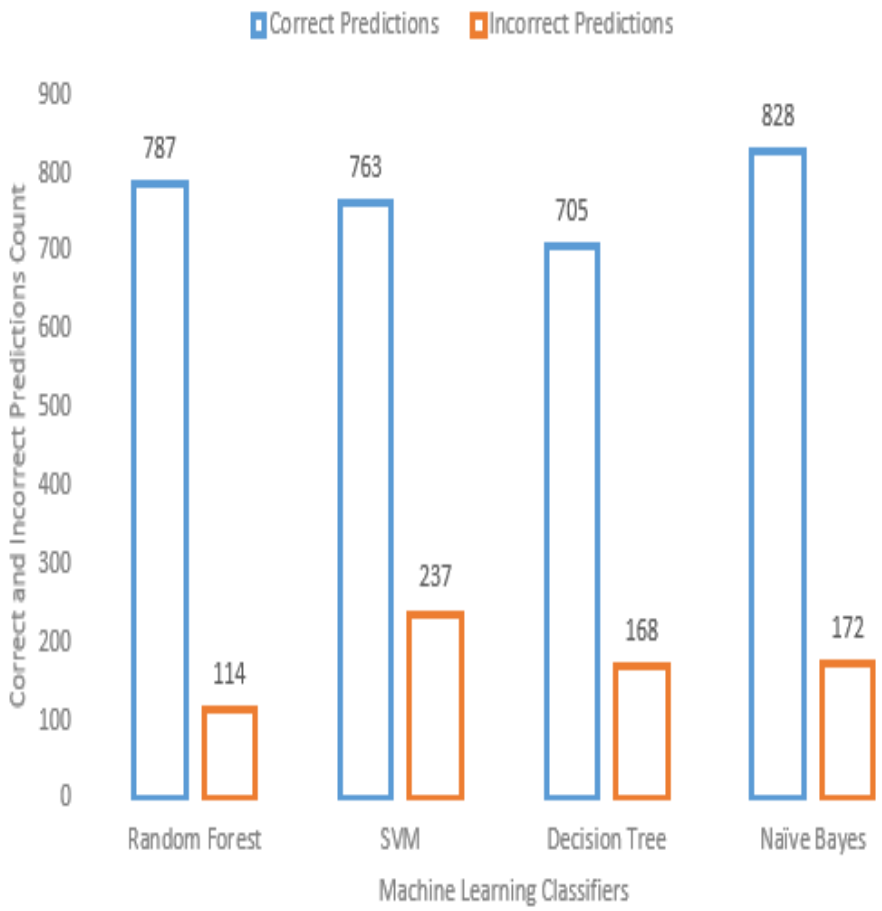


FIGURE 4.10: Total Correct and Incorrect Predictions of Classifiers

evaluation, the random forest algorithm performed well then all of the other three classifiers. SVM was the algorithm that came to the lowest position in this evaluation.

Now we compare the values of precision, recall, and f1 score of the classifiers. The value which is closest to 1 means it is a good score. Whereas the value which is near 0 means it is the worst score. Figure 7 compares the Precision, Recall, and F1 Score of all the classifiers in a graphical form.

We can see from figure 4.11 that SVM has a precision value of 0.92, Random Forest has the precision value of 0.90, Decision tree has 0.79 and Naive Bayes has a precision value of 0.83. The SVM algorithm has the best precision value as 0.92 is closest to 1 and is considered a good precision value.

Similarly, SVM has a Recall value of 0.62, Random Forest has the Recall value of 0.88, Decision tree has 0.78 and Naive Bayes has a Recall value of 0.83. The

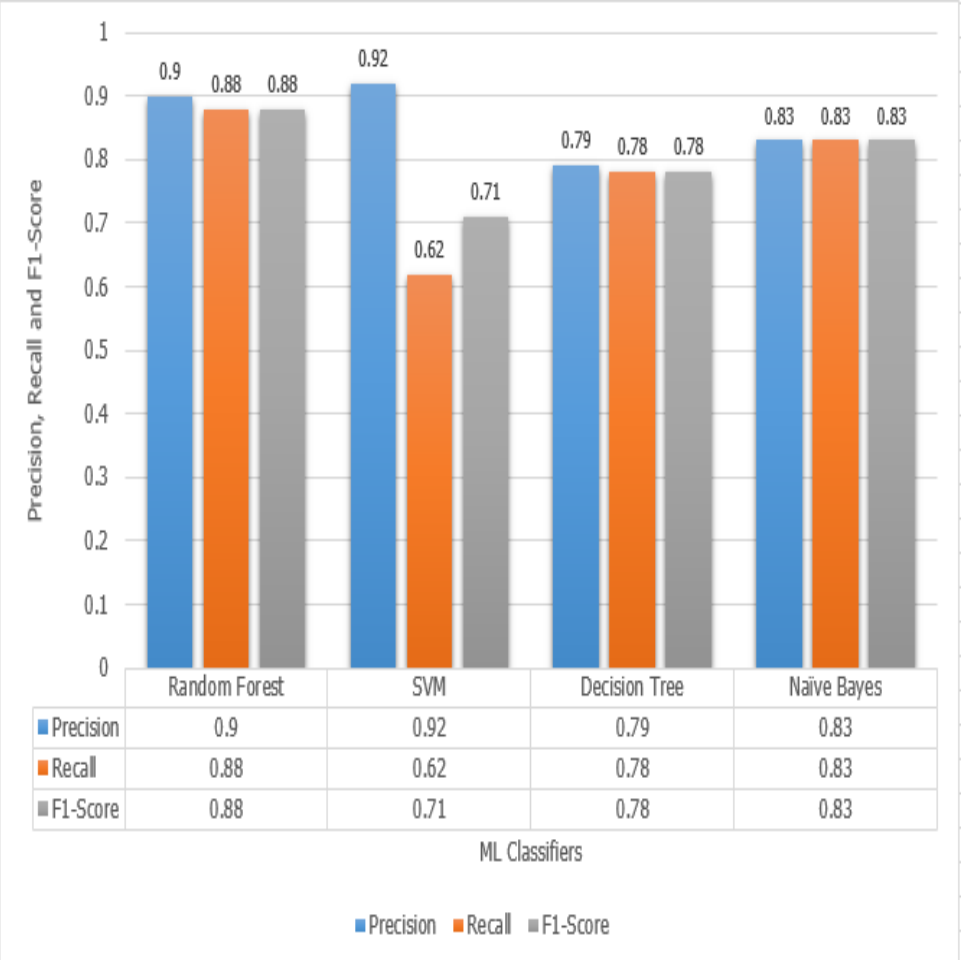


FIGURE 4.11: Precision, Recall and F1 Score of Classifiers

Random Forest algorithm has the best Recall value as 0.95 is closest to 1 and is considered a good Recall value. 1 is the perfect recall value.

When we compare the F1 Scores of the classifiers, SVM has an F1-score of 0.71, Random Forest has an F1-score of 0.88, Decision tree has 0.78 and Naive Bayes has an F1-score of 0.83. The Random Forest algorithm has the best F1-score as 0.88 is closest to 1 and is considered a good F1-score. 1 is the perfect F1-score.

TABLE 4.1: Final Evaluation - 1

	Precision	**Recall**	**F1 Score**	**Correct Predictions**
SVM	0.92	0.62	0.71	763
RF	0.90	0.88	0.88	787
J48	0.79	0.78	0.78	705
Naive	0.83	0.83	0.83	828

Table 4.3 and 4.3 shows the final evaluation of our whole experiment in a tabular format. We compared the performance of four machine learning classifiers:

Table 4.2: Final Evaluation - 2

	Incorrect Predictions	Accuracy	Time(s)
SVM	237	76.3	5.12
RF	114	88.4	3.12
J48	168	78.3	3.78
Naive	172	82.8	10.11

Random Forest, Decision Tree, Naive Bayes, and SVM. The parameters which we used for evaluation are Confusion Matrix, Accuracy, Precision, Recall, F1 Score, and model training and test time. We can see from the table that the Random Forest algorithm has performed best according to all the performance metrics we selected.

It concludes that by using the Random Forest classifier, we can find the combinations of the cryptographic algorithms on plaintext data using the type of the text, with little impact on the performance.

4.4 Summary

In this chapter, we evaluated the machine learning classifiers we implemented and compared their performances with each other using different performance metrics. We studied the literature review to find out the classifiers that are used mostly by the researchers. We used four machine learning classifiers and compared their results with each other. The name of machine learning classifiers is Support Vector Machine, Random Forest, Decision Tree, and Naive Bayes.

We compared the performances based on accuracy, confusion matrix, precision, recall, and f1 score. These evaluation metrics are mostly used by the researchers to evaluate the machine learning models. Researchers have evaluated their machine learning models using the above-mentioned metrics. Results showed that the random forest algorithm performed best in terms of metrics including model training and test time. Whereas the other algorithms we not performant enough to be declared good machine learning classifiers for our problem. That's why we recommend the random forest algorithm to recommend the combination of the

cryptographic algorithm using the plaintext type. The most In the next section, we are going to conclude our thesis.

Chapter 5

Conclusion, Limitations and Future work

5.1 Conclusions

We live in a world where everything has become data. There are millions of use cases where sensitive data is being transmitted from one place to the other over the internet of some other data transmission medium. With such an increase in data transmission, the security of the data is another important factor. It is necessary to transmit the data in such a format that it securely reaches from sender to the receiver with no modification. We need to take care of confidentiality, integrity, and availability of the data. Cryptography is used when we need to make sure that the data is safely transferred from sender to receiver. When the data is encrypted using the right cryptography algorithm, it is almost impossible for the intruder to decrypt it and read it.

Research community has been actively working in the cryptography domain to improve the quality of encryption and to make data more secure. Researchers have provided many cryptography algorithms and later on their modifications to encrypt and decrypt the data. Ensembled Cryptography is another type of cryptography where multiple algorithms are combined to use their strength and make a more strong algorithm for cryptography.

Many cryptography algorithms have also been proposed by different researchers where they have combined several cryptography algorithms together to make a more strong cryptography algorithm. In creating those cryptography algorithms, researchers have considered the block size, key size, and some other parameters to select the cryptography algorithms. But one of the important features of the data that is (plaintext type) is missing from the research. plaintext type is an important factor when selecting cryptography algorithms. As we know the stream cipher algorithms encrypt the data bit by bit and online data travels in binary format. Whereas block cipher algorithms encrypt the data in blocks and are efficient in encryption of offline nature of data or a data which is at rest. So we came up with a hypothesis to consume machine learning techniques to implement an effective machine learning model which will recommend the combination of cryptography algorithms based on the provided plaintext and its type. In our research, we limited the data to text data. Our goal was to design a machine learning model, which can recommend the combination of cryptography algorithms required to encrypt the data.

The motivation for using machine learning for implementing this technique is that machine learning has been playing an important role in classification problems for a decade in cryptography and information security as well. The nature of our experiment is also a classification one, so we used machine learning techniques to test our hypothesis and see how well machine learning can take part in our experiment.

There was no gold standard dataset available for our experiment because of our problem. As we need to train the machine learning model based on text data, its type and get the combination of cryptography algorithms in the result. So dataset creation was also an enormous challenge for us. We did a user survey with the subject experts to select the combination of cryptography algorithms. And then later on distributed the dataset between different subject experts to take their suggestions about applying the suitable combination on provided sample of plaintext and its type. After collecting all the results, we choose the results, which had most voters. Our dataset has currently 5000 rows.

After the creation of the dataset. We did research to find out which machine learning algorithms are mostly used in the domain of cryptography and network security. After doing a comprehensive literature review, we selected four machine learning algorithms to perform an experiment on and comparing their results to select the best classifier for our technique. Those were SVM, Decision Tree, Random Forest, and Naive Bayes. We used sci kit learn and python to perform our experiment. We divided the dataset into a 20:80 ratio, in which 20% was the testing dataset and 80% was the training dataset. After completing the experiment, we had to select the evaluation parameters for our classifiers evaluation and comparing those results to select the best performer.

We selected accuracy, confusion matrix, precision, recall, f1 score, and training and test time of the model to evaluate the classifiers. All the classifiers have given good scores and good accuracies, but Random Forest and Decision Tree are having almost the same accuracy and other scores except the training and test time. Random Forest seems to take a lot more time than the decision tree. So the conclusion is made that the Decision Tree seems to perform best in terms of time and quality.

5.2 Limitations

Our major limitation was the dataset available. There was no gold standard dataset available to us, so we had to create the dataset from the scratch. Currently, we had a few data available to test our proposed hypothesis. But our machine learning classifiers seemed to perform well despite data limitations.

There was a lack of research available in this area where researchers consider the plaintext to encrypt the data. There was no technique available to compare our results to. That is why we had to train multiple machine learning classifiers to compare their results.

5.3 Future work

The machine learning classifiers we implemented performed very well with the hypothesis we proposed and gave us excellent results. However, in the future, we plan to test our hypothesis on the larger dataset and other data types as well. In the current research, we are only limited to text data. We plan to implement the same technique on images and videos as well. We plan to make our dataset publicly available so that other researchers can also experiment with our data and give their suggestions.

Bibliography

[1] William Stallings. *Cryptography and Network Security: Principles and Practice*. Prentice Hall Press, USA, 6th edition, 2013. ISBN 0133354695.

[2] https://blog.mailfence.com/. Symmetric and asymmetric algorithms. https://blog.mailfence.com/symmetric-vs-asymmetric-encryption/, 2021. [Online; accessed 20-September-2021].

[3] Binay Kumar, Muzzammil Hussain, and Vijay Kumar. Brrc: A hybrid approach using block cipher and stream cipher. In Khalid Saeed, Nabendu Chaki, Bibudhendu Pati, Sambit Bakshi, and Durga Prasad Mohapatra, editors, *Progress in Advanced Computing and Intelligent Engineering*, pages 221–231, Singapore, 2018. Springer Singapore. ISBN 978-981-10-6872-0.

[4] Jignesh R.Patel, Rajesh Bansode, and Vikas Kaul. Hybrid security algorithms for data transmission using aes-des. *International Journal of Applied Information Systems*, 6:15–21, 10 2013. doi: 10.5120/ijais13-451028.

[5] Ali Taha, Dr-Diaa Salama, Salama Abdelminaam, M Khalid, and Khalid Hosny. Enhancement the security of cloud computing using hybrid cryptography algorithms. *International Journal of Advancements in Computing Technology*, 9, 12 2017.

[6] M. Harini, K. Pushpa Gowri, C. Pavithra, and M. Pradhiba Selvarani. A novel security mechanism using hybrid cryptography algorithms. In *2017 IEEE International Conference on Electrical, Instrumentation and Communication Engineering (ICEICE)*, pages 1–4, 2017. doi: 10.1109/ICEICE.2017.8191910.

[7] Vivek Kapoor and Rahul Yadav. A hybrid cryptography technique for improving network security. *International Journal of Computer Applications*, 141:25–30, 05 2016. doi: 10.5120/ijca2016909863.

[8] Chitra Biswas, Udayan Das Gupta, and Md. Mokammel Haque. An efficient algorithm for confidentiality, integrity and authentication using hybrid cryptography and steganography. In *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pages 1–5, 2019. doi: 10.1109/ECACE.2019.8679136.

[9] Lina Gong, Li Zhang, Wei Zhang, Xuhong Li, Xia Wang, and Wenwen Pan. The application of data encryption technology in computer network communication security. In *AIP Conference Proceedings*, volume 1834, page 040027. AIP Publishing LLC, 2017.

[10] Wang Tian-fu and K. Babu. Design of a hybrid cryptographic algorithm. *International Research Journal of Engineering and Technology*, 10 2012. doi: 10.1016/j.jksuci.2017.10.002-elsevier.

[11] Rawya Rizk and Yasmin Alkady. Two-phase hybrid cryptography algorithm for wireless sensor networks. *Journal of Electrical Systems and Information Technology*, 2(3):296–313, 2015. ISSN 2314-7172. doi: https://doi.org/10.1016/j.jesit.2015.11.005. URL https://www.sciencedirect.com/science/article/pii/S2314717215000616.

[12] Muhammed Al-Muhammed and Raed Zitar. –lookback random-based text encryption technique. *Journal of King Saud University - Computer and Information Sciences*, 10 2017. doi: 10.1016/j.jksuci.2017.10.002-elsevier.

[13] Ting Liu, Jue Tian, Yuhong Gui, Yang Liu, and Pengfei Liu. Sedea: State estimation-based dynamic encryption and authentication in smart grid. *IEEE Access*, 5:15682–15693, 2017.

[14] Ali Taha, Dr-Diaa Salama, Salama Abdelminaam, M Khalid, and Khalid Hosny. Improving the security of cloud computing using hybrid cryptography algorithms. *International Journal of Advancements in Computing Technology*, 9, 12 2018.

[15] Ashish Sharma, Dinesh Bhuriya, and Upendra Singh. Secure data transmission on manet by hybrid cryptography technique. In *2015 International Conference on Computer, Communication and Control (IC4)*, pages 1–6, 2015. doi: 10.1109/IC4.2015.7375688.

[16] Nishtha Mathur and Rajesh Bansode. Aes based text encryption using 12 rounds with dynamic key selection. *Procedia Computer Science*, 79: 1036–1043, 2016. ISSN 1877-0509. doi: https://doi.org/10.1016/j.procs.2016.03.131. URL https://www.sciencedirect.com/science/article/pii/S1877050916002623. Proceedings of International Conference on Communication, Computing and Virtualization (ICCCV) 2016.

[17] MP Babitha and KR Remesh Babu. Secure cloud storage using aes encryption. In *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, pages 859–864. IEEE, 2016.

[18] Nadia Mustafa Mohammed Alhag and Yasir Abdelgadir Mohamed. An enhancement of data encryption standards algorithm (des). In *2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, pages 1–6. IEEE, 2018.

[19] P. Gayathri, Syed Umar, G. Sridevi, N. Bashwanth, and Royyuru Srikanth. Hybrid cryptography for random-key generation based on ecc algorithm. *International Journal of Electrical and Computer Engineering*, 7:1293–1298, 06 2017. doi: 10.11591/ijece.v7i3.pp1293-1298.

[20] Ako Abdullah. Advanced encryption standard (aes) algorithm to encrypt and decrypt data. *International Journal on Cryptography and Information Security*, pages 222–228, 06 2017.

[21] Ünal çavuşoğlu, S. Kacar, Ahmet Zengin, and Ihsan Pehlivan. A novel hybrid encryption algorithm based on chaos and s-aes algorithm. *Nonlinear Dynamics*, 92, 06 2018. doi: 10.1007/s11071-018-4159-4.

[22] Dheerendra Mishra, Ashok Kumar Das, and Sourav Mukhopadhyay. A secure and efficient ecc-based user anonymity-preserving session initiation authentication protocol using smart card. *Peer-to-peer networking and applications*, 9(1):171–192, 2016.

[23] G. Viswanath, P. Venkata Krishna, and Sourav Mukhopadhyay. Hybrid encryption framework for securing big data storage in multi-cloud environment. *Peer-to-peer networking and applications*, 9(1):171–192, 2021.

[24] Venkata Gangireddy, Srihari Kannan, and Karthik Subburathinam. Implementation of enhanced blowfish algorithm in cloud environment. *Journal of Ambient Intelligence and Humanized Computing*, 12, 03 2021. doi: 10.1007/s12652-020-01765-x.

[25] Ashwak ALabaichi. Evaluation of a dynamic 3d s-box based on cylindrical coordinate system for blowfish algorithm. *Life Science Journal*, 15(10), 2018.

[26] Y Kumar, R Joshi, T Mandavi, S Bharti, and R Rathour. Enhancing the security of data using des algorithm along with substitution technique. *Int. J. Eng. Comput. Sci*, 5(10):18395–18398, 2016.

[27] Yong Zhang, Xueqian Li, and Wengang Hou. A fast image encryption scheme based on aes. In *2017 2nd International Conference on Image, Vision and Computing (ICIVC)*, pages 624–628. IEEE, 2017.

[28] S Arul Thileeban. Encryption of images using xor cipher. In *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pages 1–3. IEEE, 2016.

[29] Rohit K Singh, Tajunnisa Begum, Lawrence Borah, and Debabrata Samanta. Text encryption: character jumbling. In *2017 International Conference on Inventive Systems and Control (ICISC)*, pages 1–3. IEEE, 2017.

[30] Ajay Kushwaha, Hari Ram Sharma, and Asha Ambhaikar. A novel selective encryption method for securing text over mobile ad hoc network. *Procedia Computer Science*, 79:16–23, 2016.

[31] Lim Chong Han and Nor Muzlifah Mahyuddin. An implementation of caesar cipher and xor encryption technique in a secure wireless communication. In *2014 2nd international Conference on Electronic Design (ICED)*, pages 111–116. IEEE, 2014.

[32] Mohammed M Alani. Applications of machine learning in cryptography: a survey. In *Proceedings of the 3rd International Conference on cryptography, security and privacy*, pages 23–27, 2019.

[33] Sreerama K Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. *Data mining and knowledge discovery*, 2(4):345–389, 1998.

[34] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[35] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[36] Saurabh Mukherjee and Neelam Sharma. Intrusion detection using naive bayes classifier with feature reduction. *Procedia Technology*, 4:119–128, 2012.

[37] Harry Zhang. The optimality of naive bayes. *AA*, 1(2):3, 2004.

[38] Margherita Grandini, Enrico Bagli, and Giorgio Visani. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*, 2020.

[39] Kwetishe Joro Danjuma. Performance evaluation of machine learning algorithms in post-operative life expectancy in the lung cancer patients. *arXiv preprint arXiv:1504.04646*, 2015.

[40] Chih-wei Hsu, Chih-chung Chang, and Chih-Jen Lin. A practical guide to support vector classification chih-wei hsu, chih-chung chang, and chih-jen lin. *Technical Report, Department of Computer Science and Information Engineering, University of National Taiwan, Taipei*, pages 1–12, 11 2003.