A MACHINE LEARNING APPROACH TO EVALUATE THE SECURITY OF ULTRA-LIGHTWEIGHT BLOCK CIPHERS

LEE TING RONG

UNIVERSITI SAINS MALAYSIA

2021

A MACHINE LEARNING APPROACH TO EVALUATE THE SECURITY OF ULTRA-LIGHTWEIGHT BLOCK CIPHERS

by

LEE TING RONG

Thesis submitted in fulfilment of the requirements for the degree of

Master of Computer Science

October 2021

ACKNOWLEDGEMENT

This thesis would not have been a success without my project supervisor, Dr. Teh Je Sen for his great support throughout the course of this research study. He has provided me with the guidance I need to prefer a thorough background study on the relevant fields, as well as coming up the proposed solution to the problem statements. He also provided me with his helpful opinion whenever I encounter certain challenges or obstacles during the research. He also guided me on the proper format and practices to compose this report as a means to document my work. With his help, I was able to make all the correct decisions when it comes to this research.

I would also extend my gratitude to Dr. Jasy Liew Suet Yan for her support for providing me with suggestions towards improving my work as it develops, as well as her opinions as a machine learning experts as to how we should overcome certain problems during the research.

I would also like to thank my coursemates Moatsum and Wei Zhu, for giving me suggestions throughout the composition of this report, as well as pointing me towards some relevant learning materials for the study. I would also like to thank my family for providing me with the moral support I need throughout the entire project.

TABLE OF CONTENTS

CHAPTER 3 EXPERIMENTAL SETUP

CHAPTER 4 EXPERIMENTAL RESULTS AND DISCUSSION

LIST OF TABLES

Page

LIST OF FIGURES

LIST OF ABBREVIATIONS

AES Advanced Encryption Standard

ARX Addition Rotation XOR

DES Data Encryption Standard

DSA Digital Signature Algorithm

GFS Generalized Feistel Structure

kNN k-Nearest Neighbour

LEA Lightweight Encrytion Standard

MAC Message Authentication Code

MLP Multi-Layer Perceptron

MSE Mean-Squared Error

NN Neural Networks

P-layer Permutation Layer

rep_1 Permutation Representation 1

rep_2 Permutation Representation 2

RSA Ron (R)ivest, Adi (S)hamir, Leonard (A)dleman

S-box Substitution Box

SP-layer Substitution-Permutation layer

SPN Substitution-Permutation Network

TF TensorFlow

UC_1 Unseen Cipher Dataset 1 (Insecure)

UC_2 Unseen Cipher Dataset 2 (Moderately Secure)

UC_3 Unseen Cipher Dataset 3 (Secure)

LIST OF SYMBOLS

$|x|$ Bit length of x

$\oplus$ Exclusive OR

$\sum$ Sum

ENC_k Encryption with key k

DEC_k Decryption with key k

$\leq$ Less than or equal to...

$>$ Greater than...

$!$ Factorial

$||$ Concatenate

f Round / Activation Function

ΔX An XOR Difference of X

ANCANGAN PEMBELAJARAN MESIN UNTUK PENILAIAN SEKURITI SIFER BLOK RINGAN ULTRA

ABSTRAK

Kripanalisis tradisional biasanya dilakukan melalui perhitungan manual atau algorithma pencarian manual, kedua-dua kaedah yang memerlukan pengiraan yang mahal dan tidak mampu menghasilkan pandangan selain yang telah ditentukan oleh kripanalsis. Kebelakangan ini, pembelajaran mesin dianggap sebagai pendekatan yang berkesan untuk tugas-tugas pengiktirafan corak. Kebolehlenturan pembelajaran mesin telah menyebabkan aplikasinya semakin meluas dalam pelbagai bidang. Namun begitu, aplikasi and keupayaan pembelejaran mesin dalam kriptografi belum diterokai dengan teliti. Kebanyakan kerja yang wujud menggunakan pembelajaran mesin untuk meramalkan nilai-nilai kunchi rahsia, tetapi terdapat percubaan yang tidak berjaya. Kebanyakan kerja juga menyasarkan sifer-sifer tertentu dan bukan mencadangkan model yang dihasil and diseragamkan untuk tujuan generik. Untuk menyelesaikan masalah ini, karya ini bertujuan untuk meggunakan pembelajaran mesin untuk menilai keselamatan blok sifer dari perspektif kripanalisis pembezaan - standard yang diterima secara meluas ketika menilai keselamatan blok sifer. Sebaliknya meramal kunci rahsia sifer seperti kaedah-kaedah kripanalisis tradisional, model yang dilatih dalam karya ini menggunakan ciri-ciri umum daripada blok sifer. Ini merangkumi bilangan pusingan, corak permutasi dan perbezaan terpotong sedangkan label (tahap keselamatan) berdasarkan jumlah kotak penggantian aktif yang berbeza. Melalui pemilihan ciri yang dapat digeneralisasikan dan dapat diperoleh dari kebanyakan sifer blok ringan, eksperimen dilakukan dengan menggunakan pengklasifikasi pembelajaran mesin dan dioptimumkan untuk hasil yang lebih baik. Eksperimen yang dijalankan melibatkan enam pengklasifikasi (linier dan tidak linier) yang dilakukan pada sifer Feistel umum yang dipermudah sebagai bukti konsep, mencapai ketepatan ramalan sehingga 93%.

Semasa meramalkan keselamatan varian sifer yang belum dilihat, ketepatan ramalan sehingga 71% diperoleh. Penemuan ini juga menunjukkan prestasi yang lebih baik yang berasal dari pengklasifikasi tidak linier kerana sifat blok sifer yang tidak linier.

A MACHINE LEARNING APPROACH TO EVALUATE THE SECURITY OF ULTRA-LIGHTWEIGHT BLOCK CIPHERS

ABSTRACT

Traditional cryptanalysis is typically performed using manual calculations or searching algorithms, which can be computationally costly and cannot produce additional insights apart from what has been dictated by the cryptanalyst. In recent years, machine learning has been regarded as an effective approach when it comes to pattern recognition. The applications of machine learning in various areas are widespread due to its versatility. However, for cryptography in particular, it has not been explored thoroughly. Existing work leverage upon machine learning models to predict secret key values, but most attempts have been unsuccessful. Existing work has mostly targeted only specific ciphers rather than being generalisable. To solve these problems, this work aims to use machine learning classifiers to assess block cipher security from the perspective of differential cryptanalysis - a widely accepted standard when it comes to evaluating block cipher security. Rather than predicting the secret key of the cipher, the network will be trained using the general features from block ciphers. This includes the number of rounds, permutation pattern and truncated differences whereas the labels (security level) are based on the number of differentially active substitution boxes. Through the selection of features that are generalisable and obtainable from most lightweight block ciphers, experiments are conducted using machine learning classifiers and are optimised for better results. The experiments conducted involve six classifiers (linear and nonlinear) that were performed on a simplified generalised Feistel cipher as a proof-of-concept, achieving a prediction accuracy of up to 93%. When predicting the security of unseen cipher variants, prediction accuracy of up to 71% was obtained. The findings also indicate better performance coming from nonlinear classifiers due to the inherently nonlinear nature of block ciphers.

CHAPTER 1
INTRODUCTION

1.1 Background

Over the years, machine learning has received a lot of attention due to its incredible learning potential when it comes to pattern recognition and learning capabilities. Its applications are widespread and have great potential to aid researchers in their work in many fields, cryptography and cryptanalysis included.

Machine learning is the application or deployment of artificial intelligence to perform a specific task and improve upon it through progressive learning. A machine learning model would typically be tasked with predicting a specific output through formulating its own conclusions with the features presented to it. These features can be defined as vectors or variables that possess a certain relationship where the model is tasked to learn or uncover. These features are presented and computed by the model through linear or nonlinear functions that possess weights and biases. The weight of a feature can be defined as how much influence said feature has over the final output, whilst the bias offsets the weight of a feature should proper conditions are met to directly influence the output of the instance.

Cryptography can be defined as the process of writing messages in an encrypted form or code to secure said message in the face of an adversary seeking to intercept its data or contents, thus ensuring data privacy and integrity. There are different methods that seek to achieve this in the field, one of them being symmetric-key encryption. An extension of symmetric-key encryption is block ciphers, which is a well-known type of cipher that has been designed for widespread use. Lightweight block ciphers, as the name suggests, are derived from block ciphers to see more use in constrained devices with memory and processing limitations. In contrast to cryptography, cryptanalysis

is the action of attacking a cipher to obtain information of the cipher, be it from the encrypted contents, the original contents of specific features within the cipher. Ultimately, the goal of the attack is to obtain enough information to nullify the effects of a cipher.

Each cipher consists of a series of well-defined steps for the encryption and decryption processes, and these steps will contain operations that involve many predefined parameters. In the context of this research work, these parameters are referred to as cipher features. These features are either constants that are predefined by the cipher's designer such as the permutation and substitution boxes used during the encryption or decryption process, or it could also be referred to as the input/output that is decided by the users (plaintext and ciphertext). This research also deals with ciphers that are *seen* or *unseen* by a mahine learning model. A *seen* cipher would refer to ciphers whose features have been used to train machine learning models, while an *unseen* cipher would refer to cipher data that were not used to train those models.

1.2 Problem Statement

Traditional cryptanalysis is typically performed through manual calculations or by using automated searching algorithms. Thus, the capabilities and limits of these methods are directly dictated by their non-intelligent or non-dynamic nature (Chen et al., 2017; Dinur, 2014; P. Zhang & Zhang, 2018).

To improve upon these conventional approaches, machine learning algorithms such as neural networks can be leveraged. However, there is limited research on the use of machine learning in the field of cryptanalysis. Existing work typically targets a particular block cipher rather than developing standardised tools that are more applicable for general use. This can be attributed to the lack of data and the complexity in generating said data, as well as the parameters that need to be featured when working out the weights and biases. These featured parameters can be either essential information

towards the structure of the block cipher or something else of relevance that will be contributing directly to the learning rate of the neural network model that is created.

There has been existing work that highlight attempts in using machine learning models to predict secret keys (Mishra, Murthy, & Pal, 2018). However, these attempts were rather unsuccessful in their objectives, reaching a sub-50% prediction rate at best. An alternative approach needs to be identified to apply machine learning in analysing the security of ciphers. Although there have been recent advances in the field of applying machine learning in cryptanalysis, the use of cipher features apart from plaintext-ciphertext data remains largely unexplored (Gohr, 2019).

Cryptanalysis methods that are manual grow inefficient and resource consuming as the block size of the cipher increases. It is relatively easier to apply said methods towards ciphers of smaller sizes. By training a machine learning model that is tasked to predict the security margins of block ciphers, it could be possible to generalise the model to predict the security of unseen ciphers, once it is trained under the same block size. However, it remains an unknown whether if said model would be able to generalise towards other unseen ciphers with the same structure.

The following research questions will be answered in this work:

- How can a machine learning model be used to estimate the security of a block cipher?
- What are the relevant cipher features that contribute towards the security of a block cipher?
- Can a machine learning model trained using cryptanalytic data be generalised for unseen ciphers of the same size?

1.3 Research Motivation

Cryptanalysis plays a very important role to determine the security of ciphers. It also plays an important role in security evaluations to set standards that newly proposed ciphers must fulfill. Therefore, new methods of cryptanalysis need to be introduced or improved upon to ensure that the encryption standards of ciphers are up to par.

Although machine learning has proven to be useful for learning and pattern recognition tasks, the research surrounding the applications of machine learning in cryptanalysis is are mostly cipher-specific, leaving plenty of room for improvements. There is work that has proven that exploiting specific properties of a cipher - such as differentials, machine learning models can be used for conventional cryptanalytic attacks (Gohr, 2019). The research work in this thesis can also contribute towards the creation of standardised tools for evaluating the security of newly proposed ciphers.

1.4 Research Scope and Objectives

1.4.1 Research Scope

The focus of the research here is symmetric-key ciphers, specifically ultralightweight GFS ciphers. Lightweight ciphers are chosen for their performance even under constrained resources. As a proof of concept, the data used throughout this research was generated from a reduced GFS cipher. They are chosen to have a specific limit to the number of cryptographic features that can be extracted as parameters during the training process. A smaller set of features can be helpful in emphasising the importance of said features when evaluating the importance of the learning model, as well as their overall contribution to the security of the cipher.

These ciphers will be evaluated via cryptanalysis, however, rather than extracting the secret key the work conducted will create a method to evaluate the security margin - through predicting the security margin based on differential cryptanalysis. Through the machine learning model predicting the security margin of each instance through their

input and output differences, a cryptanalyst would be able to categorise the data they have to begin their research. Therefore, rather than creating a machine learning model that serves as a means to perform cryptanalysis directly, it acts as a medium assisting tool to help the cryptanalyst find a better starting point to begin their cryptanalytic attack towards their target cipher or for block cipher designers to quickly determine if their designs are secure. The research will use supervised machine learning to achieve said goal, specifically using linear and nonlinear classifiers.

1.4.2 Research Objectives

The first objective of this research is to propose a machine learning approach to predict the security of ultralightweight block ciphers. This is performed by predicting a core metric used to determine the security margins of a cipher is after being trained using relevant cipher features as input. These features include the number of s-boxes, number of rounds and the different permutation combinations. The machine learning models will be tasked to predict the security level of a block cipher based on the number of active s-boxes after providing it with cipher features as inputs. The models are tested on ultralightweight block ciphers as a proof of concept.

The second objective would be to identify cipher features that contribute towards the learning of the model and refining how they are presented in order to maximise the prediction accuracy. The main goal here is to optimise how the data could be presented in such a way that the model will have better training and testing outcomes resulting in better performance overall.

The final objective would be developing machine learning models that can be generalised to unseen ciphers. As previously explained, the data used as the input will be the features related to the structure of the cipher. Therefore, it would mean that as long as the user can provide those features as input data, a trained model can predict any cipher's security margin with respect to differential cryptanalysis as long as they share

the same size and structure.

It could be noted that all the objectives stated above are able to be measured using metrics such as accuracy, precision and recall. However, each objective has its own evaluation criteria based on accuracy.

- To propose a machine learning approach to predict the security of ultralightweight block ciphers.
- To identify features that maximise prediction accuracy and refining how they are presented.
- To develop machine learning models that can be generalised to unseen ciphers.

1.5 Research Methodology

There are five main phases for the research study conducted here. The first phase involves preliminary studies primarily on the background of lightweight GFS block cipher structures, differential cryptanalysis and the state of art for machine learning in the fields of cryptography and cryptanalysis. The second phase involves generating the dataset samples of a 4-branch GFS cipher to create training and testing datasets for the machine learning models, which are developed in the third phase of the work. The third phase includes the refinement of feature presentation as well as the configuration of hyperparameters for the machine learning models. The fourth phase involves conducting generalisation experiments that examine the accuracy of machine learning models trained with data from seen ciphers to predict the labels of data from unseen ciphers. The final phase compiles all findings of the experimental phases for documentation purposes. These phases are mapped towards the research objectives in Figure 1.1, with a more detailed explanation for each methodology phase in Chapter 3.

Steps	Summary	Related Objectives
Preliminary Studies	• Block Ciphers • Cryptanalysis • Machine Learning	Objective 2: To identify features that maximize prediction accuracy and refining how they are presented.
Generating Training and Testing Data	• Generating data • Labelling data	Objective 1: To propose a machine learning approach to predict the security of lightweight block ciphers. Objective 2: To identify features that maximize prediction accuracy and refining how they are presented.
Creation and Training of Machine Learning Model	• Creation of model • Training the model	Objective 1: To propose a machine learning approach to predict the security of lightweight block ciphers. Objective 3: To develop machine learning models that can be generalized to unseen ciphers for practical use cases.
Generalization Experiments	• Testing model accuracy • Optimization of data presentation and model	Objective 2: To identify features that maximize prediction accuracy and refining how they are presented. Objective 3: To develop machine learning models that can be generalized to unseen ciphers for practical use cases.
Compiling Findings and Results	• Collecting results • Documentation of data	Objective 1: To propose a machine learning approach to predict the security of lightweight block ciphers. Objective 2: To identify features that maximize prediction accuracy and refining how they are presented. Objective 3: To develop machine learning models that can be generalized to unseen ciphers for practical use cases.

Figure 1.1: Research Methodology

1.6 Research Contribution

Existing research has been using machine learning models to either strengthen the security of target ciphers by integrating them as part of the cipher or to behave as said cipher without having access to the algorithm altogether. They have also been used as cryptanalytic models on target ciphers. While some approaches have proven to be more successful than others, there is a lack of prior work developing generalised models that are applicable to a particular cipher structure, rather than specific ciphers. The primary and third objectives of the work conducted here will address that problem.

The proposed work introduces a new approach to predicting the security of a block cipher wherein it evaluates the security of a cipher based on its features rather than the common state of art applications where the models are primarily used as integration into the cipher or targeting the specific cipher for cryptanalysis. The integration of truncated differentials and the different data type representations in the learning process will provide a new perspective when it comes to improved cryptanalytic methods.

Therefore, the testing and training data that is used in this research will be generated from scratch as opposed to selecting an existing one through other resources. Upon completion of the research, the data used in said experiments could also be shared publicly as a contribution to future work of a similar field.

1.7 Thesis Outline

Chapter 1 has provided an overview of the research by discussing its motivations, the research problems and how the problems are addressed in the research objectives, leading to contributions towards the area of cryptography and cryptanalysis.

Chapter 2 provides an in-depth review of the background of symmetric-key block ciphers, cryptanalysis and machine learning, before going towards the current state of the art of work done in those fields. The main focus of each field would be GFS block ciphers, differential cryptanalysis and finally supervised machine learning. An introduction towards both linear and nonlinear machine learning models will also be covered here.

Chapter 3 discusses the methodology involved in this research in detail, and how each step or phase of the work is related to the main objectives of the research. Due to the novelty of the work, it will also provide information as to how the data of the research is generated and created for the experiments that will be explained in the following chapter.

Chapter 4 explains each experiment phase of the research on how to set up and replicate the process in detail. The results and findings of each phase will also be presented here.

Chapter 5 concludes the work conducted throughout this research and provides a discussion on its advantages, disadvantages and limitations. Future work will also be discussed.

CHAPTER 2

LITERATURE REVIEW

As explained prior in the problem statement and research contributions, existing work that was conducted using machine or deep learning models either integrated them as part of the cipher, or used them for predicting or extracting information (usually secret key bits) from ciphers. This chapter will cover these existing research works along with background information related to both machine learning and cryptography. Sections 2.1 to 2.4 cover background information related to this field whereby 2.1 to 2.3 introduces cryptography and cryptanalysis, followed by Section 2.4 covering machine learning and its various models. Section 2.5 and 2.6 will be covering the state-of-art applications of machine learning in the fields of cryptography and cryptanalysis.

2.1 Cryptography

The purpose behind cryptography is to be able to communicate securely in the face of an adversary. This field has grown by a large degree over the past few decades. Cryptography can be broken down into two main categories as seen in Figure 2.1. One encryption, and the other authentication. Authentication is the process where one party tries to verify the identity of the other party inside a secure channel. For instance, the server will verify who the client is before executing the client's requests, all the while the client tries to verify if the server is whoever it claims to be. It is important to note that authentication merely verifies the identity of involved parties, and is not particularly involved in encrypting the message. Cryptographic hash functions and message authentication codes (MAC) are commonly used methods for authentication and ensuring data integrity.

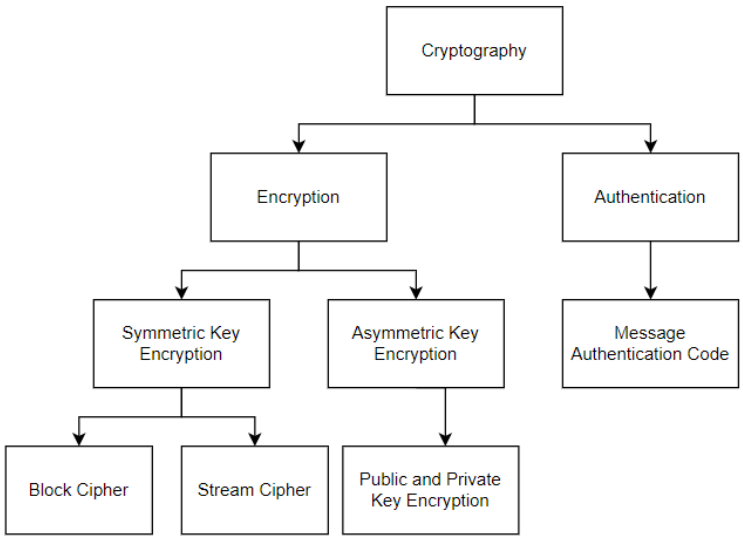

Figure 2.1: Taxonomy of Cryptography

Encryption is a key concept in cryptography whereby a message is encoded into an unreadable format so that it confuses eavesdropping people. The encrypted texts, or better known as ciphertexts, will then be sent through a secure channel to its recipient and the ciphertexts are decrypted to the original plain text.

Symmetric key ciphers are ciphers that are algorithms for cryptography that use the same cryptographic keys for both encryption and decryption of its contents. The key can be either a number, a word, a string of random letters, or a combination of the lot. It is meant to be used to change the contents of the texts in a way so that the recipient can decrypt the messages provided the sender and recipient share the knowledge of the key. Symmetric key ciphers are mainly divided into two categories, namely block ciphers and stream ciphers. A block cipher is an encryption algorithm that encrypts a fixed size of data, otherwise known as a block (hence the name), at a time. Further discussion of block ciphers can be found in the section below. Some examples of widely used block cipher algorithms are PRESENT (Bogdanov et al., 2007), Camellia (Aoki et al., 2001) and AES (Daemen & Rijmen, 2002).

A stream cipher, however, is a cipher that encrypts 1 bit or byte(s) of plaintext at a time. Stream ciphers often use pseudorandom bits as the key that are generated unpredictably. In a stream cipher, the plaintext digit/bit is encrypted one at a time with the corresponding digit/bit in the keystream before combining them to become the ciphertext. A popular stream cipher example would be SNOW (Ekdahl & Johansson, 2003).

Asymmetric key encryption, also known as public-key cryptography uses two keys to encrypt a plaintext. The structure emphasises the importance of its two-key structure, where the public key is made freely available to anyone, whilst the second private key is known only to its owner. The public and private keys share a mathematical relationship such that a message that has been encrypted using a public key can only be decrypted using a private key. The primary benefit of the structure is users are not required to disclose their private keys to reduce the chances of having information about it being intercepted by the adversaries during transmission. Well-known asymmetric key cipher techniques include RSA (Rivest, Shamir, & Adleman, 1983) and DSA.

2.2 Block Ciphers

Block ciphers take b-bit message and k-bit key as an input, then processes it with a key-dependent transformation and outputs a string of the same number of bits. It is a well-known type of cipher that has seen widespread use. A block cipher has two essential parameters: block size (denoted as b) and the key size (denoted as k).

The same secret key will lead to the same set of round keys. In order to encrypt multiple blocks of data, block ciphers take b-bit string blocks and outputs respective b-bit string blocks under transformations that are dependent on these round keys. The message encryption and decryption can be represented by the formula below:

$$\begin{aligned} c_i &= ENC_k(m_i) \\ m_i &= DEC_k(c_i) \\ for 1 &\leq i \leq n \end{aligned} \tag{2.1}$$

where c represents the ciphertext and m represents the plaintext and i is the block index of the message.

Blocks ciphers can be divided into several main categories based on their internal structures. Popular block cipher structures include substitution-permutation networks (SPN), generalised Feistel structures (GFS) and addition-rotation-XOR (ARX). SPNs process plaintext through a series of interleaved substitution and permutation operations that are repeated for multiple rounds. For GFS ciphers, the nonlinear operations (round functions) are performed on only half of the input, and data permutation is performed on sub-blocks rather than individual bits. ARX uses a combination of modular addition, rotation and XORs to perform linear and nonlinear operations. It is important to note that substitution boxes (s-boxes) are not involved and instead ARX uses the aforementioned operations to increase its resilience against linear and differential cryptanalysis. Examples of ARX ciphers include SIMON, SPECK and LEA (Beaulieu et al., 2015; Hong et al., 2014). More information about these cipher categories is provided in the upcoming subsections.

2.2.1 Confusion and Diffusion

Block ciphers are designed in the sense that they can provide enough of both confusion and diffusion to make sure adversaries are unable to decrypt their messages. Confusion ensures that the ciphertext is statistically dependent on the plaintext in a

manner too complex to be exploited by the adversary or cryptanalyst. Diffusion ensures that each bit of the plaintext, alongside the key, influences multiple bits of the ciphertext.

Both confusion and diffusion are generally implemented through a well-defined series of substitution and permutation operations. Substitution is responsible for confusion, replacing bits in a plaintext with new ones by following a certain rule, whereas permutation is responsible for diffusion by manipulating the order of the bits using a certain algorithm. It helps to diffuse plaintext and key bits throughout the ciphertext.

2.2.2 Substitution and Permutation

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	2	c	4	1	7	a	b	6	8	5	3	f	d	0	e	9
p_1	e	b	2	c	4	7	d	1	5	0	f	a	3	9	8	6
p_2	4	2	1	b	a	d	7	8	f	9	c	5	6	3	0	e
p_3	b	8	c	7	1	e	2	d	6	f	0	9	a	4	5	3

Figure 2.2: Sample S-box (Knudsen & Robshaw, 2011)

Substitutions are used to provide the confusion property in a cipher and are typically implemented as a lookup table known as an s-box. To minimise the memory requirements of s-boxes, the size of s-boxes are typically 4 or 8 bits. Figure 2.2 is an example of an s-box. For example, 9 will be substituted as 5 if p_0 substitution box is used, whereas when the p_1 is used, 9 will be substituted as 0.

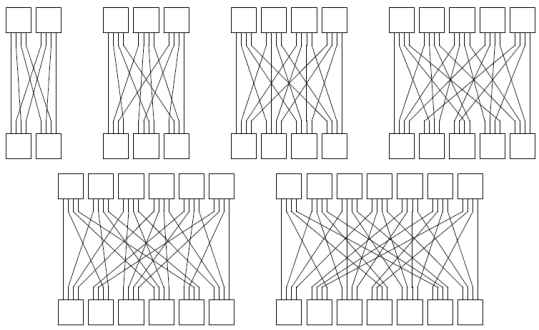

Figure 2.3: Sample Permutation Layer (Leander, 2010)

Permutation contributes to the diffusion, shuffling bits or nibbles in ciphertext blocks in a specified fixed manner. It fulfills the diffusion role by changing the location of various bits to diffuse the effect of round key or plaintext bits throughout the ciphertext. Figure 2.3 above are example permutation patterns for 8-bit permutations all the way up to 56-bit permutations.

A typical formula to calculate different combinations of a text under the use of permutation is seen as:

$$P(n,r) = \frac{n!}{(n-r)!} \tag{2.2}$$

Where n represents the number of possible elements for each bit of the message, and r represents the number of total bits in the message. Both substitution and permutation when combined serves as an important component for block cipher design, otherwise known as the Substitution Permutation Networks or SPN.

2.2.3 Substitution-Permutation Networks: SPN

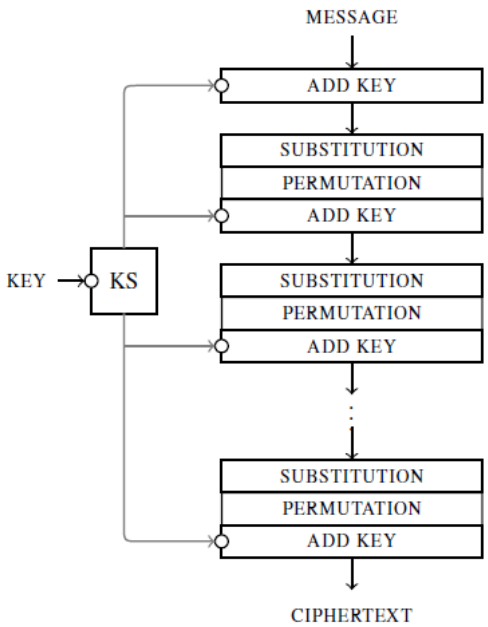

Figure 2.4: Sample SPN (Knudsen & Robshaw, 2011)

SPN takes the plaintext as an input and applies alternating rounds or layers of s-

boxes and permutation layers (P-layers) to produce the final ciphertext which is showcased on the Figure 2.4. Decryption is performed by inverting the entire process, which requires an inverse s-box and an inverse permutation pattern.

2.2.4 Feistel Ciphers and Generalised Feistel Structures (GFS)

Feistel ciphers are another example of symmetric key block cipher where it consists of the repetition of r rounds of an identical structure, with each round consisting of a round function and a swap. The round function maps a $b/2$-bit input to a $b/2$-bit output under the action of a set of round keys.

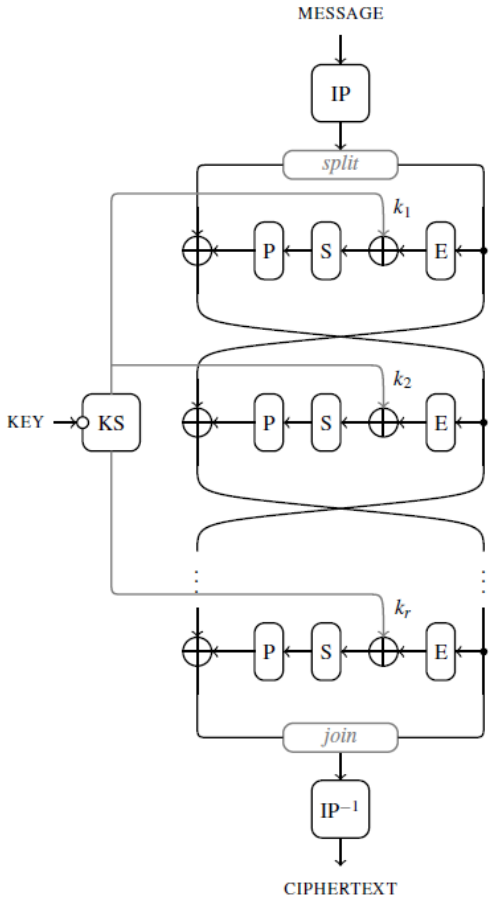

Figure 2.5: Feistel Cipher (Knudsen & Robshaw, 2011)

The message inputs can be written as:

$$m = (u_0||v_0) \tag{2.3}$$

where u_0 and v_0 are both 32-bits (for the case of a 64-bit cipher) and the || represents their concatenation. The ciphertext output is written as:

$$c = (u_r||v_r) \tag{2.4}$$

and the intermediate values u_i and v_i are generated where $f(\dot{})$ represents the round function:

$$\begin{aligned} v_i &= u_{i-1} \oplus f(v_{i-1}, k_i) \\ u_i &= v_{i-1} \\ u_r &= u_{r-1} \oplus f(v_{r-1}, k_r) \\ v_r &= v_{r-1} \\ where\ 1 &\leq i \leq r-1 \end{aligned} \tag{2.5}$$

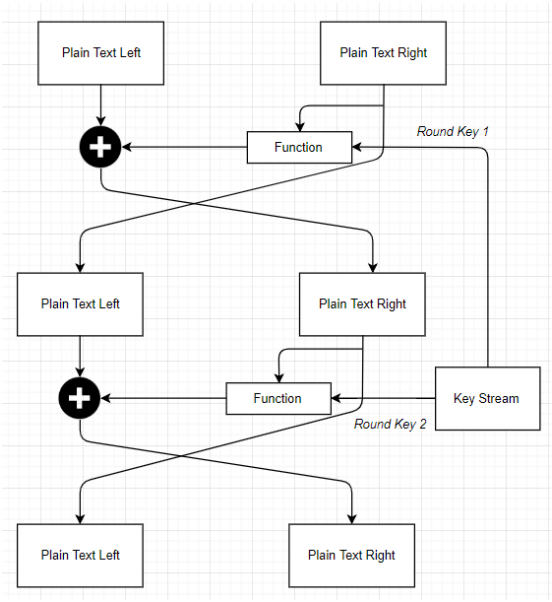

Figure 2.6: General flow of Round Functions

Round functions are a repeated process that happens between Feistel ciphers with the involvement of round keys. As seen from Figure 2.5, the round function is responsible for mapping a $b/2$-bit input into a $b/2$-bit output under the action of the round keys $k_1 k_r$. The output will be used to modify one half of the text that is being encrypted, before swapping both halves and proceed to the next round. This will continue until the last round in the cipher, where there is no swap and the halves will only be joined to provide the output. The example form of the integration of a round function into a cipher can be seen on Figure 2.6.

The advantage of this network is that the encryption and decryption processes can use the same structure. The only difference is that the order of rounds keys need to be reversed to obtain the decrypted text.

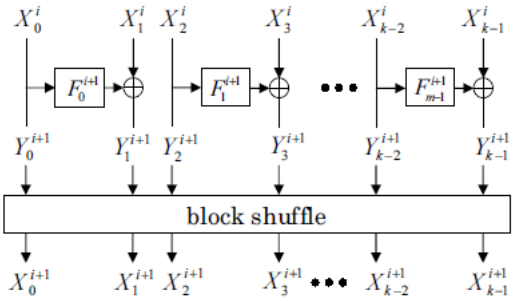

Figure 2.7: Sample GFS cipher Suzaki and Minematsu (2010)

Generalised Feistel structure (GFS) is the generalisation of the classical Feistel cipher. There are different versions of GFS but they all share the similarity of dividing their message into k sub-blocks where $k > 2$, commonly referred to as the partition number. Figure 2.7 provides a sample operation of what is happening in a round of a GFS, with i being the current number of rounds and F representing the round key function of the said round. Well-known GFS ciphers include TWINE and LBlock (Suzaki, Minematsu, Morioka, & Kobayashi, 2013; Wu & Zhang, 2011).

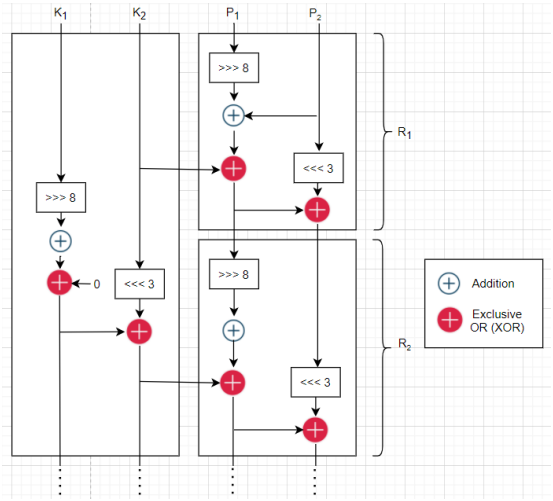

Figure 2.8: ARX Cipher

2.2.5 ARX

ARX ciphers are another class of symmetry key cipher algorithm that mainly consists of three important operations: additions modulo 2^n, bit rotations and XORs. Figure 2.8 shows a sample operations that is carried out in each round of the cipher - shifting bits of the text or key schedule, addition and XOR of key and plaintext pairs. The biggest strength of ARX is the fact that it uses these linear and nonlinear operations for the encryption rather than lookup tables like s-boxes. Common examples of ARX ciphers include SPECK (Beaulieu et al., 2015) and LEA (Hong et al., 2014).

2.2.6 Lightweight Block Ciphers

Lightweight ciphers are proposed for constrained devices, taking into consideration memory and processing limitations. Its overall goal for both hardware and software implementations are to also keep the resource consumed during the encryption and decryption processes to a bare minimum, without compromising the security of the overall cipher. This includes the time required for each cycle, memory restrictions as well as the frequency of each cycle.

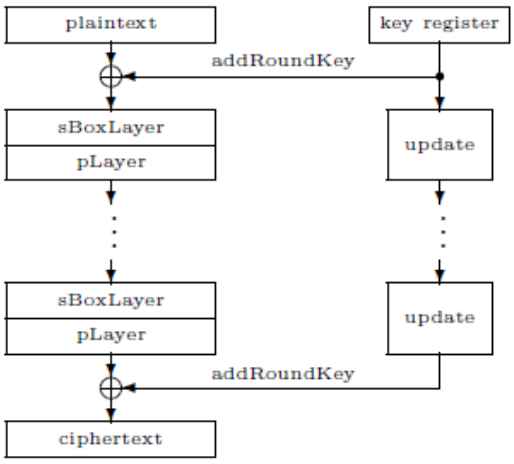

Figure 2.9: PRESENT Cipher (Bogdanov et al., 2007)

PRESENT is an example of a lightweight SPN cipher consisting of 31 rounds. As shown in Figure 2.9, each round consist of an XOR operation to introduce a round key K, for $1 \leq I \leq 32$, *where* K_{32} is used for linear bitwise permutation and a non-linear substitution layer, as well as post-whitening which involves XOR-ing the data with parts of the key. Besides keeping security and efficiency in implementation, the main goal when designing PRESENT was simplicity. It is designed with an extremely constrained environment in mind, and therefore it is important to note that the main purpose of the cipher was not to be one for widespread use (as AES has already covered that front), but to provide one that is lighter on resource consumption such as total space and time consumed; all the while providing security levels that are optimal for a smaller key size.

2.3 Cryptanalysis

Cryptanalysis is the action of attacking a cipher to obtain information from either the plaintext, ciphertext or key. In most cases, the attacker or adversary is trying to recover the secret key of the cipher. The outcome of cryptanalysis can be divided into several broad categories such as:

- Total Break: Adversary obtains the key.

- Global Deduction: The adversary is able to discover an algorithm that is equivalent to the encryption or decryption process of the cipher.
- Local Deduction: Adversary can generate the plaintext corresponding to a previously unseen ciphertext, and vice versa.
- Distinguishing Algorithm: The adversary is able to distinguish between two black boxes, one being the cipher and the other being a randomly chosen permutation.

Cryptanalytic attacks can be further classified based on what information is available to an adversary. These are also known as attack models. These classifications (in descending order of difficulty) include:

- Ciphertext Only: Able to intercept and obtain the ciphertext
- Known Plaintext: Able to intercept and obtain the ciphertext and is known that said ciphertext corresponds to a certain plaintext.
- Chosen Plaintext: Adversary creates a number of plaintext for the cipher to encrypt, then intercepts the ciphertext equivalents.
- Chosen Ciphertext: Adversary chooses the arbitrary ciphertext for the cipher, then intercepts the plaintext decrypted from it.
- Open Key Model: Adversary has prior knowledge of the key of the target cipher. (So, 2020)
- Side-Channel: Not strictly a cryptanalytic attack. Uses other data about the encryption or decryption process to gain information of the message, such as electronic noise produced by encryption machines and sounds produced by keystrokes when typing the plaintext. (Jap & Breier, 2014; Levina, Sleptsova, & Zaitsev, 2016)

While there are various cryptanalysis methods available, this research places more emphasis on differential and its counterpart, linear cryptanalysis, both of which rely on the concept of active s-boxes.

2.3.1 Linear Cryptanalysis

Linear cryptanalysis takes advantage of high probability occurrences of linear expressions involving plaintext bits and ciphertext bits from a nonlinear operation. The basic idea is to approximate the operation of a portion of the cipher with a linear expression such that it refers to a mod-2-bit wise operation, such as XOR. This approach requires the identification of linear expressions with either a high or low probability of occurrence, which is evidence of biases that can be exploited in key recovery attacks (Heys, 2002). Linear cryptanalysis has been applied to various block ciphers, proving itself to be a viable method of cryptanalysis (Biryukov & De Canniere, 2005; Bogdanov & Rijmen, 2011).

2.3.2 Differential Cryptanalysis

Differential cryptanalysis is a chosen-plaintext attack in which adversaries are assumed to have gained access to the encryption algorithm and can obtain ciphertexts based on selected plaintexts. Differential cryptanalysis requires the identification of high probability differential trails constructed by the chaining of interconnected differences.

A differential is a difference propagation from an input difference to an output difference, while a differential characteristic specifies that as well as the internal difference after each round. The effects of a secret key will be cancelled out once the difference is considered. The results of its applications have proven that differential cryptanalysis is effective in its role (Biham, Dunkelman, & Keller, 2006; Dunkelman & Keller, 2008; Lu, Dunkelman, Keller, & Kim, 2008; Lu, Kim, Keller, & Dunkelman, 2008; Tsunoo et al., n.d.).

A difference between a pair of data blocks, $X' = [X'_0, X'_1, ..., X'_{i-1}]$ and $X" = [X"_0, X"_1, ..., X"_{i-1}]$ is defined as:

$$\Delta X = X' \oplus X" \tag{2.6}$$

$$\Delta X = [\Delta X_0, \Delta X_1, ..., \Delta X_{i-1}], \tag{2.7}$$

where X' and X'' represent the pair of input blocks of the block cipher and Y' and Y'' are their corresponding output blocks. The pair, $\{\Delta X, \Delta Y\}$ is known as a differential pair. For an ideal cipher, given any particular input difference ΔX, the probability of any particular ΔY occurring will be exactly $\frac{1}{2^b}$ where b is the number of bits. Differential cryptanalysis relies on the existence of a differential, $\Delta X \rightarrow \Delta Y$ with a probability far greater than $\frac{1}{2^b}$.

2.4 Machine Learning

Machine learning is an application of artificial intelligence that provides systems with the ability to automatically learn and improve from experience without explicit programming. They emphasise the development of computer programs in a way that said programs can access data and learn for themselves. The machine learning algorithms typically try to look for patterns in data and make better decisions in future sets of data based on their deducted findings. This is extremely important as the primary aim of machine learning is to be able to have the computer learn automatically by itself without intervention or assistance. There are two main categories that classify machine learning algorithms: supervised and unsupervised learning.

2.4.1 Supervised Learning

Supervised machine learning is the use of machine learning algorithms to produce a general hypothesis based on externally supplied instances of data, which will then make improvements on said hypothesis for future instances of data to produce better or more accurate outputs. The purpose of supervised machine learning is to build a

model the is capable of predicting a class or label of a data instance based on prediction features, typically known as regression or classification problems. Models are designed to either classify and categorise the data based on output and observation (classification), or inferring the dependencies and relationships between the variables (Kotsiantis, 2007).

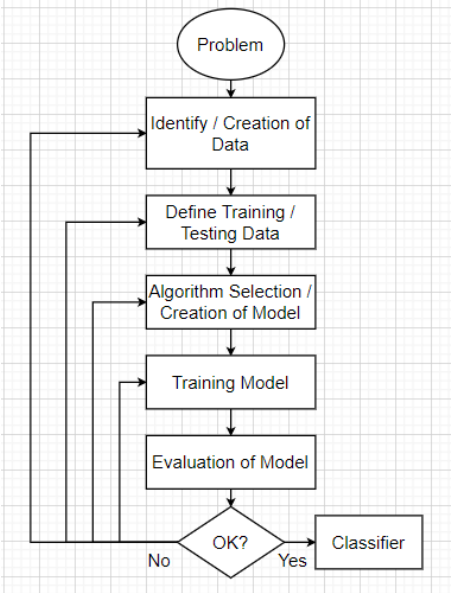

Figure 2.10: Classification Technique (Kotsiantis, 2007)

Classification is a technique where the model is typically tasked to determine what group the set of data belongs to. Seemingly like how a human can learn to differentiate plants and animals over time, machine learning algorithms can do the same. As shown in Figure 2.10, the model will apply a classification algorithm to identify the shared characteristics of certain classes, compare those characteristics to the current data, then uses their deductions to estimate and classify what category said data is tied to. Regression, on the other hand, is a method of modelling a target value based on independent predictors. This is mostly used in the works of forecasting and finding out the relationship between the variables involved and how their interactions will affect the outcome. It has the goal to have the algorithm predict the real-valued output.

2.4.2 Unsupervised Learning

Unlike supervised learning, unsupervised learning algorithms infer patterns from a dataset without reference to known outcomes. This means that instead of the classification role of supervised learning, unsupervised learning is responsible for the class discovery. Common examples of unsupervised machine learning include clustering and partitioning. Clustering starts by breaking down a single cluster of data into smaller clusters after every iteration before grouping the object that is closest among each other into smaller clusters and repeating the process. Partitioning involves separating the data into smaller clusters based on a specified final number of clusters before assigning samples to each partition. The samples will then attract remaining samples that are related to them hence “partitioning" the data Gentleman and Carey (2008).

2.4.3 Linear Classifier

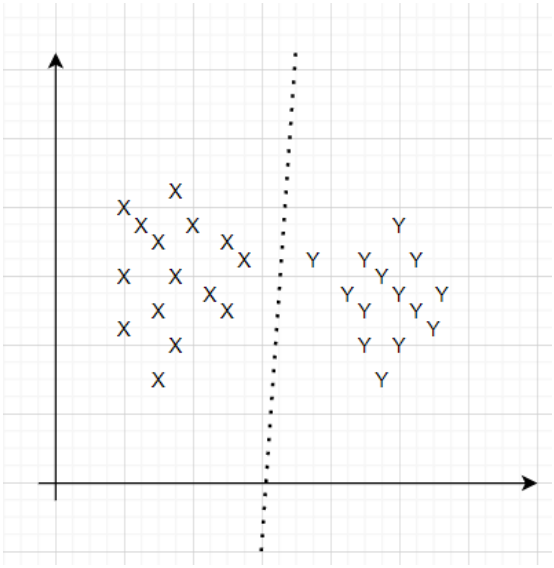

Figure 2.11: Linear Classification Problem

Linear classifiers solve classification tasks based on a linear combination of features. The Figure 2.11 shows X and Y as sample classes, the goal of linear classifiers is to segregate, as accurately as possible, the training data into their respective classes using a linear function (i.e., a straight line).