UNIVERSIDAD DE GRANADA

TRABAJO FIN DE MASTER

INGENIERÍA INFORMÁTICA

Cryptanalysis Ciphertext Based Genetic Algorithms

Breaking Transposition and Substitution Ciphers with Genetic Algorithm

Autor

Abdullah Taher Saadoon AL Muswai (alumno)

Directores

Juan Julián Merelo Guervós (tutor)

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

Granada, septiembre de 2020

Cryptanalysis Ciphertext Based Genetic Algorithms

Breaking Transposition and Substitution Ciphers with Genetic Algorithm.

Autor
Abdullah Taher Saadoon AL Muswai (alumno)

Directores
Juan Julián Merelo Guervós (tutor)

Cryptanalysis Ciphertext Based Genetic Algorithms: Breaking Transposition and Substitution Ciphers with Genetic Algorithm

Abdullah, AL Musawi(student)

Keywords: Cryptanalysis, Genetic Algorithm, Transposition Cipher, substitution cipher, decryption ,optimization search.

Abstract

This thesis describes a method of deciphering messages encrypted with transposition cipher and substitution cipher utilising a Genetic Algorithm to search the keyspace. There are many tools used in the cryptanalysis, Genetic algorithm(GA) is an optimization search tool to find the best solution.

In this project, A

fitness measure based on **English Letter Frequencies** for random of text is described which got it by generating random popultaion. The results are compared to those given using a previously published technique and found to be superior.

Yo, **Abdullah Taher Saadoon AL Muswai**, alumno de la titulación máster de Ingeniería Informática de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI A11518169, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Master en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: Abdullah Taher Saadoon AL Muswai

Granada a 10 de septimpre de 2020.

D. **Juan Julián Merelo Guervos (tutor)**, Profesor del Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada.

Informan:

Que el presente trabajo, titulado ***Cryptanalysis Ciphertext Based Genetic Algorithms,Breaking Transposition and Substitution Ciphers with Genetic Algorithm***, ha sido realizado bajo su supervisión por **Abdullah Taher Saadoon AL Muswai (alumno)**, y autorizamos la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expiden y firman el presente informe en Granada a X de mes de 201 .

Los directores:

Juan Julián Merelo Guervos (tutor)

Agradecimientos

Poner aquí agradecimientos...

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introduction

Cryptanalysis is the technique of deriving the original message from the cipher text without any prior knowledge of secret key or the erivation of key from the cipher text. A general technique for cryptanalysis, applied to all cryptographic algos is to try all the possible keys until the correct key is matched, it is known as exhaustive key search.

With every passing day, the computing ability of hardware is increasing manifold; therefore it becomes necessary to use long keys for avoiding exhaustive key search. All the other attacks applied to stream ciphers are compared to exhaustive key search in terms of data and memory complexity and if its complexity is less than exhaustive key search, then only these are considered as successful.

A symmetric key cipher, especially a stream cipher is assumed secure, if the computational capability required for breaking the cipher by best-known attack is greater than or equal to exhaustive key search. Cryptanalysis is the science of making encrypted data unencrypted use convert cipher text to plaintext because cryptanalysis used to convert plaintext to cipher text and used cryptanalysis Return to plaintext or clear text or original text cryptanalysis is used to break codes by finding weaknesses. There are many techniques used in the cryptanalysis. This project used the genetics algorithm [1].

The genetic algorithm is a search algorithm based on the mechanics of natural selection and natural genetics. The genetic algorithm belongs to the family of evolutionary algorithms, along with genetic programming, Evolution strategies and evolutionary programming. The set of operators usually consists of mutation, crossover and selection [2].

A genetic algorithm has proven to be reliable and powerful optimization technique in a wide variety of applications. It can be applied to both texts and images. Genetic algorithm is secure since it does not utilize the natural numbers directly. The genetic algorithm used for generating keys that it should be good in terms of coefficient of autocorrelation [3].

Apply the technique of genetic algorithms to the problem of finding the key to a particular Transposition Cipher. Since Genetic Algorithms are primarily used to efficiently search a large problem space we thought they would be ideally suited for searching the large key space [4].

1.1. aim of the project

The main purpose of the project is examine the possible applications of genetic algorithms in cryptology,with the emphasis of the research being in the application of a genetic algorithm in the which explores the plaintext from cipher text based on genetic algorithm (GA) which is used for suggesting decryption key. The genetic algorithm (GA) is a search tool to insure high probability of finding a solution by decreasing the amount of time in the key space searching. this project covers two ciphers type: Transposition Cipher and Substitution Cipher. We present a review of Transposition Cipher ,Substitution Cipher, English Letter Frequencies and genetic algorithm in chapters ***** which provides enough background to understand the techniques applied and to assess the usefulness of the results obtained.

1.2. project organization

In additional to chapter one (the introduction), this project is organized into three other chapters: -Chapter * **"Theoretical Background"** this chapter introduces terms, definitions, and descriptions that are used frequently throughout this chapter. These terms focus on Cryptanalysis, Transposition Cipher, and description Genetic algorithm (GA). Chapter Three **"Design and Implementation: Cryptanalysis based on Genetic Algorithm"** this chapter shows the design of the project and explains the algorithms that support this work. Chapter Four **Çonclusions and Future Work"** this chapter concludes this project and finding the outline future works in the area of Cryptanalysis.

Capítulo 2

Planning

This chapter presents the work plan to follow to carry out this study. First of all, the requirements necessary to achieve the proposed objective will be captured, and then the work planning will be detailed, with an estimate of both work and material costs as well as a distribution of time between tasks.

2.1. Investigation stages

The steps of the research process of this study are as follows:

1. **Initiate a preliminary investigation of the Cryptanalysis:** review many of theses and papers looking for information about Cryptanalysis, check what evolution they have followed over the years and how the Cryptanalysis systems have been developed.

2. **delve into the Transposition and Substitution ciphers:** investigate further in these types that are really going to focus on the study

3. **Select the algorithm:** investigated in the previous steps, deciding the most appropriate option.

4. **delve into genetic algorithm:** Study and understand all stages of genetic algorithm evolution and how to solve current problems, review the papers that related with GA.

5. **Implement genetic algorithm:** create a set of classes, each of them performs stage of GA stages and all of them has a set of methods to divide its stage to small tasks every task solves a different problem.

6. **create a new proposal for the study:** for the objective of the study, new proposals that can compete with the state of the art must be proposed.

7. **Implement my new proposal:** that will be part of the experiment have been decided, they must be implemented in the same way as the state of the art algorithm. Check of new that all the functionality of the experiment is valid with the new implementation.

8. **Obtain the results of the experiment:** obtain through the implementation of the experiment, can visualize the data in tables and graphs that are suitable for the study.

9. **analysis of the results obtained:** compare the results of each method with the *baseline* reference method and with the state of the art method, as well as between them, to discover which method behaves best for each case.

10. **Consider future work to be done:** assess the aspects it is possible to continue progressing in the methods proposed in the study, propose new tasks or challenges within this field of research, that have not been covered in this study.

2.2. Work plan

This section is subdivided into cost estimation and time estimation. In the first place, The cost of the infrastructure used is evaluated, as long as the material that has been used is available for free and free. From there we can make a base budget from which to start. The time estimate should be realistic and meet the requirements within a competent time frame so that you do not need to postpone times at the last minute due to poor planning. If so, it would be necessary to bear the consequent increase in costs.

2.2.1. Cost estimate of materials and infrastructure

First of all, the computer with which this study has been developed will be taken, the following hardware and software have been used for this study: **Computer Type:**Lenovo IdeaPad Z510. **Processor:**Intel(R) Core(TM) i7-4702MQ CPU@ 2.20GHz 2.20GHz. **Installed memory (RAM):** 8.00 GB. **graphics card:** Intel(R) HD Graphics 4600, Memory:2176 MB. **Hard Disk:** HDD 1TB

Operating system: Win10. **Programming language:** java. **Environment:** NetBeans 8.1. All the research, implementation, and evaluation

tasks have been carried out with this team: documentation, review of the literature, writing of the report, development, implementation of the algorithms, and execution of the experiment to obtain results.

information and documentation required for the work are available at the University of Granada Thanks to the agreements that the University establishes with some documentary databases such as Scopus, it has been possible to access a multitude of papers and theses on the subject to be investigated. Furthermore, the Google scholer has been very useful to access other papers not found in Scopus.

The development of the work has been carried out entirely on the Win10 OS, The main software tools used have been open source so they have not entailed additional cost, Netbeans as the main IDE for development and writing the report of the project in LaTeX using the TeXstudio, available for win 10, and we used Visual Studio Code as an IDE to deal with the repository of the project that has hosted on **GitHub** [5] platform, and we used **Travis CI** [6] to host continuous integration service used to build and test software projects hosted at GitHub,Travis CI provides various paid plan for private projects, and a free plan for open source. Of all this material exposed in the previous paragraphs, the work has only required buying the personal laptop and the broadband network available, The rest of the infrastructure and materials have either been free software products or have been contributed by the University of Granada.

2.2.2. Time distribution to the tasks

we used the concepts of iterations and sprints to distribute the work throughout the weeks, in which a series of tasks have been developed that they fulfill a specific objective.

The Iteration and Sprint Planning meeting is for team members to plan and agree on the stories or backlog items they are confident they can complete during the sprint and identify the detailed tasks and tests for delivery and acceptance

The planning that has been foreseen for the development of the project is as follows:

Weeks	10
Estimated total time (h)	240h
date of Starting Point	01-06-2020
date of end Point	10-09-2020

2.2.3. Preparatory phase

In this phase, the development of the project will begin, which has an estimated effort of 40 hours. A feasibility study of the project idea will be carried out, an investigation about the possible tools to use

Iteration 1: Analysis and study of the project.

week	1
Total expected time (h)	40h
date of Starting Point	01-06-2020
date of end Point	06-06-2020

Sprint 1: Description of the Project

Product	Description	Week	Time
Objectives	Description of the objectives.	1	1.5
functionalities	Description of functionalities.	1	3
Motivations	Personal motivations around the project.	1	1.5
Programming	Description of possible programming methods for the proposed project.	1	4

Sprint 2: Research and preparation of the environment.

Product	Description	Week	Time(h)
Study and analysis of market	Search for similar projects and study on their components	1	10
Technologies and tools	Study of technologies and tools necessary for the development of the project	1	15
Work environment	Creation of the GitHub repository and development environment of the project	1	8

2.2.4. Implementation and testing phase

the objective of this phase is getting an general idea of this project and at the end of this phase, a software-implemented incrementally ha-

ve been obtained, where each module has been tested until integration and validation. A total of 280 hours has been planned for this phase, and it has been distributed in the following iterations:

Iteration 2: Implementation of the main classes.

The objective of this iteration is the implementation of the main classes which will create the system,For this operation, the following schedule has been established:

week	6
Total expected time (h)	120h
date of Starting Point	07-06-2020
date of end Point	18-07-2020

Product	Description	Week	Time(h)
information	Study needed: java libraries, fitness measure, Crossover operators ….	6	10
Population class	Populate the initial population with completely random solutions	6	10
Transposition and Substitution classes	Try to decode the cipher texts.	6	25
Fitness class	Calculate the fitness equation which is using English letter frequencies	6	20
Crossover class	Apply Crossover operators on population.	6	30
Mutation class	Apply Mutation operator on population.	6	5
Testing	Implementation of the tests unit of the classes developed in this iteration.	6	20

Iteration 3: Experimental results and Improvement. The objecti-

ve of this iteration is trying to get a result from the classes then try to improve all of them to get better results,For this iteration, the following schedule has been established:

week	10
Total expected time (h)	120h
date of Starting Point	18-07-2020
date of end Point	18-08-2020

Product	Description	Week	Time(h)
information	Review papers that related for each class and compare the results with.	10	15
Population class	repopulate populations with different cases	10	10
Transposition and Substitution classes	Trying to improve these classes to enhance run time	10	25
Fitness class	Compare fitness results before and after enhancing and adding the tables of English letter frequencies	10	25
Crossover class	Try to find new crossover operators and compare the result with previous result.	10	25
Mutation class	Try to balance mutation ratio with each population.	10	10
Testing	Implementation of the tests unit of the classes developed in this iteration.	10	10

2.2.5. Documentation phase

After the project is completed and approved, it is time to document all of the steps and how to work, and document the achieving result.

the documentations have been divided into two parts. The first one corresponds to the documentation in the Github repository. This documentation has as objective to give an idea about the project in general then delve in details until arrives in programming steps that explain all of the class and their methods and variables, Anyone who accesses to the repository, he can see all this information as a free software project.

Other documents made are those for the project. In this case, more documentation focused on the scope of the project will be implemented: planning, research, development, Conclusions.

Iteration 4: GitHub Documentation. In this phase has an estimated effort of 50 hours.

week	11
Total expected time (h)	50h
date of Starting Point	18-08-2020
date of end Point	24-08-2020

Product	Description	Week	Time(h)
ABSTRACT and Introduction	Insert ABSTRACT and Introduction of the project in README File	11	5
Implementation	Add Implementation of all the part of the project.	11	15
Programming	Add description of all the class of the project and explain them step by step.	11	20
Experimental results	Put the Experimental results for each class in README	11	10

Iteration 5: Project documentation.

In this phase has an estimated effort of 150 hours.

week	13
Total expected time (h)	150h
date of Starting Point	24-08-2020
date of end Point	09-09-2020

Product	Description	Week	Time(h)
Information	Review documentation about Latex	13	5
Create LaTeX file	Generation of template, packages, .abstract and cover page	13	3
Introduction	Writ all part of introduction	13	10
Planning	Writ all part of the planning chapter	13	20
Chapter 3			
Chapter 4			
Chapter 5			
Chapter 6			
Chapter 7			

Capítulo 3

Theory Background

3.1. Introduction

Cryptanalysis is the technique of extracting useful information about the key by observing the plaintext and cipher text using cryptanalysis try to break the secrecy provided by the cipher. There is no fixed method for cryptanalysis and every cipher is a different challenge to the attacker and hence demands different insight to attack[7],The study of cipher text in an attempt to restore the message to plaintext is known as cryptanalysis. Cryptanalysis is equally mathematically challenging and complex as cryptography. Because of the complexity involved with cryptanalysis work this document is only focused on the basic techniques needed to decipher monoalphabetic encryption ciphers and cryptograms[8]. In this chapter, explained the history of cryptanalysis, the technology of cryptanalysis, transposition cipher, substitution cipher and description genetics algorithm (GA).

3.2. Cryptanalysis

Cryptanalysis is the technique of deriving the original message from the ciphertext without any prior knowledge of secret key or derivation of key from the ciphertext. A general technique for cryptanalysis, applicable to all cryptographic algorithms is to try all the possible keys until the correct key is matched, it is known as exhaustive key search. With every passing day, the computing ability of hardware is increasing manifold; therefore it becomes necessary to use long keys for avoiding exhaustive key search [1] and till today, many cryptanalytic attacks are developed based on these. Each variant of these have different methods

to find distinguisher and based on the distinguisher [9].

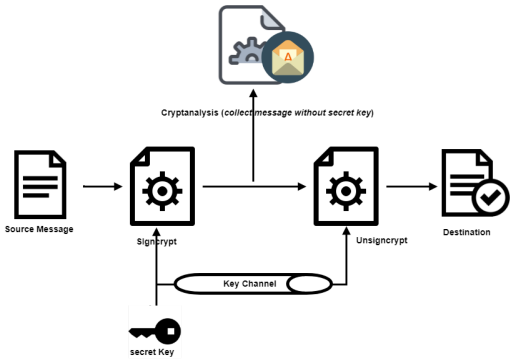

Figura 3.1: Cryptanalysis cipher text

3.2.1. CLASSIFICATION OF ATTACKS

The main goal of a cryptanalyst is to obtain maximum information about the plaintext (original data).Classification of attacks can be done on following basis [10]:

1. **Cipher text only attacks:** This is the most powerful attack. The attacker has only the knowledge of cipher text. This type of attack is successful only on the weakest of the ciphers.

2. **Known plaintext attack:**attacker has the knowledge of plaintext and the corresponding cipher text, e.g. if an attacker is eavesdropping then he can also guess the plaintext corresponding to some cipher texts depending upon the position or state of communication, in other words, In this type a cryptanalyst have plaintext and their corresponding cipher text . Attacker tries to find out the relation between these two.

3. **Chosen plaintext:**the attacker can choose its plaintext and get the cipher text corresponding to those chosen cipher text or The attacker obtain the various ciphertext corresponding to an arbitrary set of plaintext.

4. **Chosen cipher text:** attacker is able to get the decrypted plaintext corresponding to his choice of cipher text. This attack is same as the chosen plaintext, but in a reverse direction which means The attacker obtain the various plaintext corresponding to an arbitrary set of cipher text

5. **Adaptive chosen plaintext:**attacker first observes a large number of cipher texts. Based on the distribution of the cipher texts the attacker chooses a plaintext to get the corresponding cipher text which means the attacker chooses subsequent set of plaintext which is based on the information obtain from previous encryption methods.

6. **Related Key:** This is a relatively new attack model. Here the attacker can encrypt two plaintext (same plaintext or the two plaintexts with a constant difference) with two keys, which have a fixed relation Between each other. This attack model is very weak as there is very little chance for the attacker to get encryption with two keys with a Constant relation. For lightweight block ciphers as the key is written to the device, this type of attack is not very probable.

3.2.2. Cryptanalytic technique

In this section we will explain various cryptanalytic technique. As said earlier that, there are no fixed methods for cryptanalytic techniques for any block ciphers. But there are some methods which can be applied to every ciphers with some variation, though there can not be any guarantee that these methods may break the cipher. Cipher designers apply these methods to analyze security level for the computational security. Informally, broadly classify these techniques as brute force techniques non-brute force techniques. As the name suggest brute force techniques involves search of entire key space. Other techniques utilize the weakness in the structure of the ciphers to find key bits [7].

Brute force technique:

A brute-force attack is a can be used to attempt to decrypt any encrypted data. Such an attack might be used when it is not possible to take advantage of other weaknesses in an encryption system (if any exist) that would make the task easier. When password guessing, this method is very fast when used to check all short passwords, but for longer passwords other methods such as the dictionary attack are used because a brute-force search takes too long. Longer passwords, passphrases and keys have more possible values, making them exponentially more difficult to crack than shorter ones. can be made less effective by obfuscating the data to be encoded making it more difficult for an attacker to recognize when the code has been cracked or by making the attacker do more work to test each guess [7].

Non-brute force techniques:

In a non-brute-force attack, a single (usually common) password is tested against multiple usernames or encrypted files. The process may be repeated for a select few passwords. In such a strategy, the attacker is Generally not targeting a specific user. Non brute-force attacks can be mitigated by establishing a password policy that disallows common passwords [7].

3.3. Transposition Cipher

The transposition cipher is rearranged (change position only) the characters in the message but not change the characters. Transposition

cipher have a pool of keys and ciphertext that rearranged the ciphertext for M times depended on the pool of keys. The output of transposition cipher saved in array of M locations we can called it *plaintextArray*.

A simple transposition or permutation cipher works by breaking a message into fixed size blocks, and then permuting the characters within each block according to a fixed permutation, say P. The key to the transposition cipher is simply the permutation P. So, the transposition cipher has the property that the encrypted message contains all the characters that were in the plaintext message. In other words, the unigram statistics for the message are unchanged by the encryption process. The size of the permutation is known as the period. Let's consider an example of a transposition cipher with a period of ten 10, and a key P=7,10,4,2,8,1,5,9,6,3. In this case, the message is broken into blocks of ten characters, and after encryption the seventh character in the block will be moved to position 1, the tenth moved character in the block will be moved to position 2, the forth is moved to position 3, the second to position 4, the eighth to position 5, the first to position 6, the fifth to the position 7, the ninth to the position 8, the sixth to the position 9 and the third to position 10.

In Table **??** shows the key and the encryption process of the previously described transposition cipher. It can be noticed that the random string "X"was appended to the end of the message to enforce a message length, which is a multiple of the block size.It is also clear that thedecryption can be achieved by following the same process as encryption using the ïnverse.ºf the encryption permutation. In this case the decryption key, P-1 is equal to 6,4,10,3,7,9,1,5, 8,2.

KEY:
Plaintext: 1 2 3 4 5 6 7 8 9 ***10***
Ciphertext:7 ***10*** 4 2 8 1 5 9 6 3
ENCRYPTION:
Position : 123456789***10*** 1234 5678 9*10* 123456789***10***
Plaintext : TRANSPOSITION _ ALGORITHMXXXXXXX
Ciphertext OTNRSTSIPAGI _ OOIARLNXXXHXTXXXM

Cuadro 3.1: Transposition Cipher

Bibliografía

[1] Mohammad Ubaidullah Bokhari, Shadab Alam, and Faheem Syeed Masoodi. Cryptanalysis techniques for stream cipher: a survey. *International Journal of Computer Applications*, 60(9), 2012.

[2] Salvatore Mangano. Genetic algorithms. *Computer Design*, 1995.

[3] K Sindhuja and S Pramela Devi. A symmetric key encryption technique using genetic algorithm. *international journal of computer science and information technologies*, 5(1):414–416, 2014.

[4] Jason Brownbridge. Decrypting substitution ciphers with genetic algorithms. *Department of Computer Science. University of Cape Town*, page 12, 2007.

[5] repository of the project in github. `https://github.com/AbdullahTaher93/TFM`.

[6] testing of the repository of the project in github using Travis CI. `https://travis-ci.org/github/AbdullahTaher93/TFM`.

[7] Vikash Kumar Jha. Cryptanalysis of lightweight block ciphers. *Aalto University School of Science Degree Programme of Computer Science and Engineering, Master's Thesis*, 2011.

[8] Craig Smith. Basic cryptanalysis techniques. *November 17th*, 2001.

[9] Mehak Khurana and Meena Kumari. Variants of differential and linear cryptanalysis. *IACR Cryptol. ePrint Arch.*, 2015:473, 2015.

[10] William Stallings. *Cryptography and network security, 4/E*. Pearson Education India, 2006.