Unit 1 Notes

Ankur Mishra

2/28/2018

Contents

1 Columnar Transposition Ciphers

1.1 Simple Columnar Transposition Cipher

General Process: Rearranging letters based on numeric key.

1.1.1 Encoding

To encode, put text in to a matrix, which has c columns, where c = key. Then copy each column from left to right to create your a new encoded message.

1.1.2 Decoding

To decode, first approximate the number of rows (r) by dividing the length of the decoded message by the key. Then write each letter of the encoded message from up to down, till the row length is reached. Keep in mind, if there is a reminder(x) when calculating the number of rows, leave x spaces blank in the last row.

1.1.3 Example

Text: "REARRANGING LETTERS" | Key: 5
$18/5 = 4R2$

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ \\ R & E & A & R & R \\ A & N & G & I & N \\ G & L & E & T & T \\ E & R & S & & \end{bmatrix} \tag{1}$$

Encoded: RAGEENLRAGESRIT RNT

1.2 Keyword Columnar Transposition Cipher

General Process: Rearranging letters based on a keyword and each keyword letter's alphabetical order.

1.2.1 Encoding

To encode, put text in to a matrix, which has c columns, where c = the length of the keyword. Keep each letter of your keyword above each column

of the matrix. Then copy each column from based on your keyword's alphabetical order to create your a new encoded message.

1.2.2 Decoding

To decode, first find the length of the ciphered text, then divide it by the length of the keyword, which will return the number of rows r. If it has a remainder x, round the number of rows up and you will have x blank spaces in the text. Fill each column out based on the alphabetial order of the keyword and by copying every r letters from the ciphered text into the column. Finally copy each row and you will get your plaintext back.

1.2.3 Example

Text: "SLIGHTLY MORE COMPLICATED" | Keyword: CRYPTO

$$\begin{bmatrix} C & R & Y & P & T & O \\ \\ S & L & I & G & H & T \\ L & Y & M & O & R & E \\ C & O & M & P & L & I \\ C & A & T & E & D & \end{bmatrix} \tag{2}$$

Encoded: SLCCTEI GOPELYOAHRLDIMMT

2 Playfair Cipher and Railfence Cipher

2.1 Playfair Cipher

2.1.1 Encoding

To encode, first create a 5x5 matrix with first the keyword and then followed by the rest of the letters of the alphabet, while omitting any later repeats in the phrase. Then use these three rules to encode the plaintext:

different rows and columns	$\rightleftharpoons$
only same row	$\rightarrow\rightarrow$
only same col	$\Downarrow$ $\Downarrow$

If two of the same letters occur consecutively, add a space character such as X or Z. Loop to the begining or end if at edges.

2.1.2 Decoding with Cribs

It's a shaft. Have fun!

2.2 Railfence Cipher

2.2.1 Encoding

Write the text in three rows like a railfence. Then copy each row, which will create your ciphered text.

$$\begin{bmatrix} T & & & C & & & E & \\ & H & & S & & I & & H & & R \\ & & I & & & P & & & \end{bmatrix} \tag{3}$$

Encoded: TCEHSIHRIP

2.2.2 Decoding

To decode, round, divide the length of the ciphered text by 2 and ignore any remainders. This will be the length of the 2nd row, and the first and third row will be half of it. Then copy it down accordingly to create the fence, which can be rewritten as the plaintext.

3 ADFGVX Cipher and Vigenere Cipher

3.1 ADFGVX Cipher

3.1.1 Encoding

To encode, put it through a ADFGVX which is formatted like this: $//$ and find the row and column letter corresponding to it, which will be A, D, F, G, V, or X. Then encode this with Keyword Columnar Transform, which will result in the encoded message.

3.1.2 Decoding

First part to decoding is to decode the Keyword Columnar Transform. Then find the letter corresponding to every two consecutive letters which will be its row and column. To this process over the entire ciphertext and then you are done.

3.2 Vigenere Cipher

3.2.1 Encoding

To encode, either attain or make a keyword. Then find the index of each letter of the plaintext and the keyword and the sum of the two mod 26 will result in your ciphered letter.

3.2.2 Decoding

To decode, attain keyword. Then find the index of each letter of the ciphertext and the keyword and the difference between the two will result in your plaintext letter.

3.2.3 Friedman Test

3.2.4 Kasiki Test

1. Find the most common trigraphs in the ciphertext and the difference between them
2. The GCD of all the differences is the length of the keyword.

3.2.5 Steps to Decrypt

1. Use Friedman and Kasiki Test to attain length of your keyword n.
2. Split your ciphertext in to n cosets.
3. For each coset, compare their relative frequencies in the cosets to their relative frequencies in the coset. The one's with smallest differences between the two are likely to be the letters for the keyword.
4. Based on this form your keyword and then decode.