CS641A (Modern Cryptology)

Chapter-3 Report

Instructor: Prof. Manindra Agrawal

Group Name: CMEN

Names :	Pratyush	170500
	Pravar Deep Singh	160508
	Tushar Garg	160749

Date of submission of report : 1/02/2020

In Chapter 3, as we enter we see the following message:

The chamber is completely dark. You quickly pull out the match box and light a stick ...

The light fills up the chamber slowly. By now you are used to dim lights and so see things immediately. The chamber is, like the previous ones, made by carving through the rocks. Its floor is somewhat uneven, but there are no boulders here. There seems to be a constant rumbling sound in the background. You could see some odd shapes lying on the floor in a corner. Becoming curious, you move towards them and all of a sudden, freeze in your tracks. These are human skeletons!! One of them has both its hands (whatever is left of it) pointing upwards as if pleading something. The thought strikes your mind that perhaps these people could not get past the chamber and just died! Clearing your mind of negative thoughts, you hastily withdraw and start looking around. you realize that there is another door in one side of the chamber (the same side where the skeleton s lie, in your panic you did not notice this earlier). And the door is not closed! Perhaps there is another chamber here. You decide to investigate ...

Now we exhaustively tried all the commands which we got by typing the keyword "**list**" which we got from the mail, and we get the following message by using the command "**enter**":

This is a small chamber. The rumbling sound become louder here. The floor is full of small rocks. There is a stale, and somewhat bad, odour here. There is a small door to your left from which you entered. You

throw the partially burnt matchstick down and light another one.

You notice a large hole in the ground. Next to it, there is a very small hole in the ground, barely enough to put your hands in it.

Since "some message" is written in the text we tried the command "**enter**" and obtained the following message:

The rumbling sound is very loud here. It is very stickly and smelly too. You want to quickly get out of this place. Steeling yourself, you begin to investigate. It is a very small opening. The floor is muddy. You see some mushrooms growing out of the floor.

Now we picked some mushrooms using the command "**pluck**" and then we entered "**back**" and entered the command "**give**" so that the mushroom were eaten by a spirit and the magic words "**thrnxxtzy**" were revealed. We came back to the main chamber and speak the magic words "**thrnxxtzy**" and reached the following message:

A door appears in front the front wall! So does a glass panel next to it!!

We then entered the command "**read**" to read the message written on glass panel and obtained the following message:

aetypec dy ubir kcvg kehh ea aheieiv au bwe imeltpi
zwkyuy keivkkon ou kbe chbe. nt bcett, ovv dovk tgwc td
aetypknu obe zeihh ob ckx tiru au bwe ekhs qtdtdy. uei
bzkkuy cd beu rtex sto it khgwit gkbw ucm. kodv uex
btnrg ktou bvtu khgh huw ecm muc cd beu rteis. ku cmghv
tpxe wmt c xtkrnkto, co heiu ibtq toddyu! tc nu bcycnbm,
iztpe ubz teiicygv:

ceo_iqlxv_yi

We realized that the message is encrypted and we need to decrypt it.

Getting To The Encryption Method:

To find out what encryption method might be used, we first did the frequency analysis of the ciphertext. The result is shown in the graph below. The frequency distribution of normal English text is also shown below. Further, we calculated the Index of coincidence of the given ciphertext and found it to be **0.0573 ~ 0.06,** which is very close to the standard English text's Index of coincidence value of **0.0667**. Therefore, we could see high possibilities of monoalphabetic **substitution cipher** or **permutation cipher** or a combination of both.

Now by looking at the frequency graph, we realized that the frequency of letters like 'u', 'k', 'c' are very high, therefore we ruled out the case of simple permutation cipher (since only permuting won't suddenly increase relative frequency of letters). Therefore, we concluded that the encryption should be a combination of permutation and substitution.

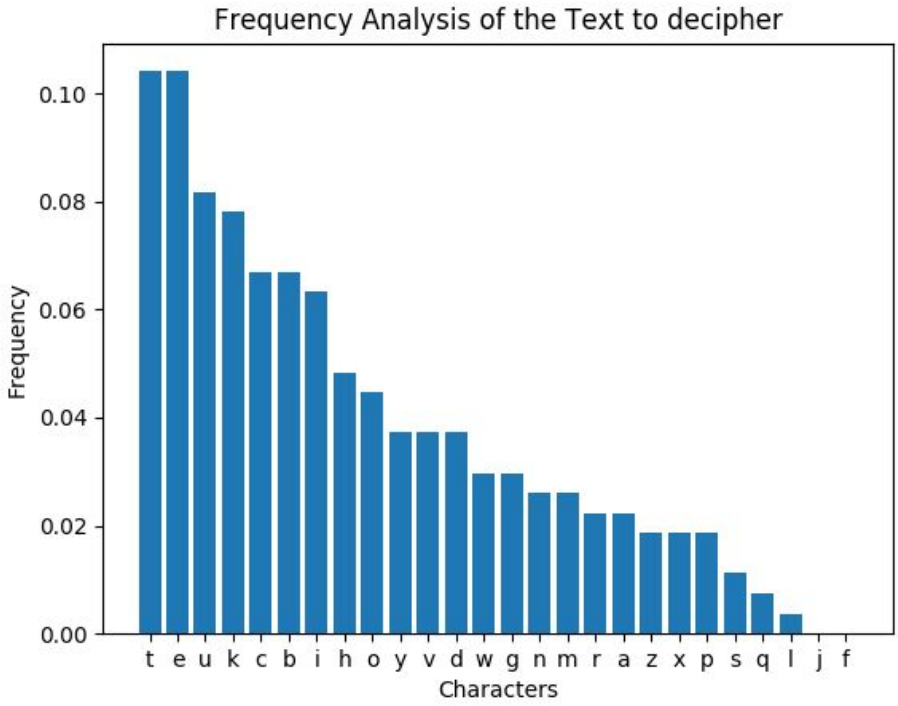

Fig. 1: Frequency Analysis of text to decrypt

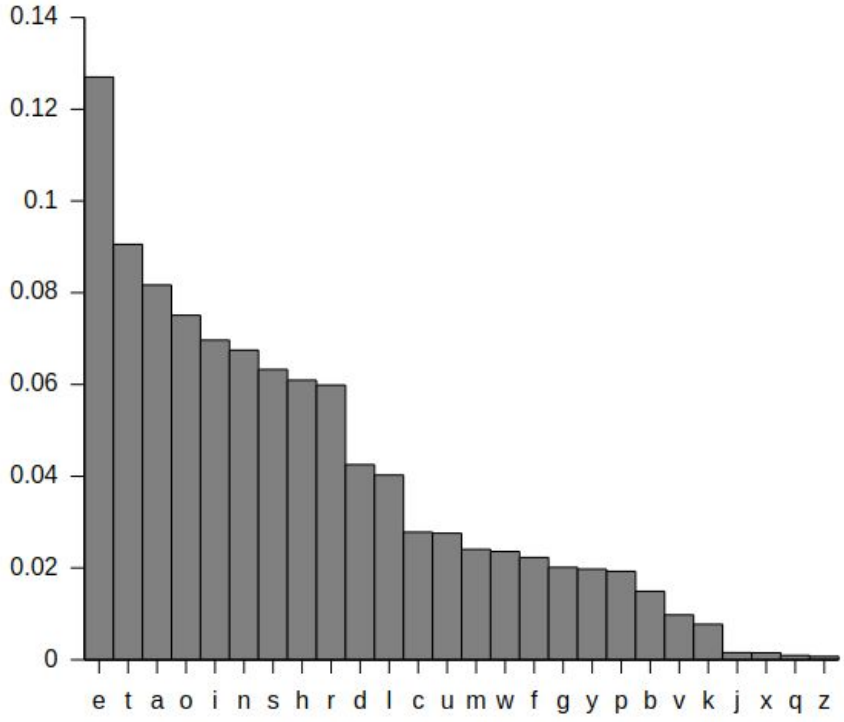

Fig. 2: Frequency Analysis of the Normal English Text

Breaking The Ciphertext:-

The number of alphabets in the encrypted message is 280 (270 paragraph + 10 passwords length). Now since block length should be a factor of the total number of alphabets, we have the following two cases:-

a. The password and paragraph are encrypted together in that case the block length is a factor of 280 i.e. 1, 2, 4, 5, 7, 8, 10, 14, 20, 28, 35, 40, 56, 70, 140 or 280.
b. The password and paragraph are encrypted separately, in that case, the block length is a factor of 270 and 10 both i.e. 1, 2, 5 or 10.

So the possible candidates for block length are 1, 2, 4, 5, 7, 8, 10, 14, 20, 28, 35, 40 and 56.

Now, in general, there are very high chances of repetition of words in English texts. Also, our cipher contained a significant amount of three-lettered words. Now words like “the”, “and”, “for”, “you” etc. are pretty common and have high chances of being repeated in the original plaintext (particularly “the”). So if we undo the permutation of the ciphertext, then the ciphertext thus formed (which has only substitution cipher left) will have a significant repetition of words (since monoalphabetic substitution won’t affect the number of repetitions).

Now let us start with the case of smaller block lengths. For smaller block lengths we can iterate over all permutations (in legible time) and check for what permutation the repetition of words is maximum (for larger block lengths we will have to guess a few words and then work accordingly. For e.g. we could guess that last word (which is 8 lettered) can be “password”).

So we first started with the assumption that block length is one of 1, 2, 4, 5, 7, 8 or 10. We wrote a code (file name permutations.py) which for each block length mentioned above, takes all possible permutations corresponding to that block length and applies that to the ciphertext and counts the number of repeated words. We found that maximum repetitions are 28 corresponding to the block length 10, σ_1: (2, 3, 4) (7, 8, 9) and block length 5, σ_2: (2, 3, 4) (written in cycle notation). Here both permutations are essentially the same (σ_1 is nothing but σ_2

repeated twice). Therefore, we concluded that it is most likely that block length is 5 and substituted text can be obtained from ciphertext by applying σ: (2, 3, 4) on the ciphertext.

Now after permuting the ciphertext according to this permutation we obtained the following text:

ayetpey cd ubki rcve gkhh ae aheiiev aw ube ilmetpw
izkyku yeikvkon ko ube bche. nc tbetv, tov dkov tgtw cd
ayetpkon ube izehh co bkx rtiu aw ube eskh qtddty. ube
izkyku cd ube rtse xto ki thgtwi gkub wcm. dkov ube
xtnkr gtov ubtu gkhh heu wcm cmu cd ube rtsei. ku gcmhv
xtpe wcm t xtnkrkto, oc heii ubto qtddty! uc nc ubycmnb,
izetp ube ztiigcyv:

Now since the above passage is a meaningful English text with monoalphabetic substitution cipher, we know that the single-letter word ‘t’ should correspond to either ‘a’ or ‘i’ but from frequency analysis, we know that the letter ‘t’ has too high frequency to be ‘i’ so it should correspond to ‘a’.

Now we had two highest frequency letters ‘e’ and ‘t’ out of which ‘t’ is ‘a’ so ‘e’ should correspond to ‘e’.

Now the word ‘ube’ appears 9 times in the text so it should correspond to the word ‘the’, which also agrees with our previous deduction of ‘e’ being ‘e’. So now the text looks like this:

ayEApEy cd THki rcvE gkhh aE ahEiiEv aw THE ilmEApw
izkykT yEikvkon ko THE HchE. nc AHEAv, Aov dkov AgAw cd
ayEApkon THE izEhh co Hkx rAiT aw THE Eskh qAddAy. THE
izkykT cd THE rAsE xAo ki AhgAwi gkTH wcm. dkov THE
xAnkr gAov THAT gkhh hET wcm cmT cd THE rAsEi. kT gcmhv
xApE wcm A xAnkrkAo, oc hEii THAo qAddAy! Tc nc THycmnH,
izEAp THE zAiigcyv:

Now the above text contains two letter words ‘Tc’ and ‘kT’ which should correspond to ‘TO’ and ‘IT’. Substituting whatever we found till now gives

ayEApEy Od THIi rOvE gIhh aE ahEiiEv aw THE ilmEApw izIyIT yEiIvIon Io THE HOhE. nO AHEAv, Aov dIov AgAw Od ayEApIon THE izEhh Oo HIx rAiT aw THE EsIh qAddAy. THE izIyIT Od THE rAsE xAo Ii AhgAwi gITH wOm. dIov THE xAnIr gAov THAT gIhh hET wOm OmT Od THE rAsEi. IT gOmhv xApE wOm A xAnIrIAo, oO hEii THAo qAddAy! TO nO THyOmnH, izEAp THE zAiigOyv:

Now we have “THIi” and “THAo” which should match to ‘THIS’ and ‘THAN’. Substituting we get

ayEApEy Od THIS rOvE gIhh aE ahESSEv aw THE SlmEApw SzIyIT yESIvINn IN THE HOhE. nO AHEAv, ANv dINv AgAw Od ayEApINn THE SzEhh ON HIx rAST aw THE EsIh qAddAy. THE SzIyIT Od THE rAsE xAN IS AhgAwS gITH wOm. dINv THE xAnIr gANv THAT gIhh hET wOm OmT Od THE rAsES. IT gOmhv xApE wOm A xAnIrIAN, NO hESS THAN qAddAy! TO nO THyOmnH, SzEAp THE zASSgOyv:

Now we get “OmT’ and ‘NO hESS THAN” which should correspond to “OUT” and “NO LESS THAN” correspondingly. Substituting we get

ayEApEy Od THIS rOvE gILL aE aLESSEv aw THE SlUEApw SzIyIT yESIvINn IN THE HOLE. nO AHEAv, ANv dINv AgAw Od ayEApINn THE SzELL ON HIx rAST aw THE EsIL qAddAy. THE SzIyIT Od THE rAsE xAN IS ALgAwS gITH wOU. dINv THE xAnIr gANv THAT gILL LET wOU OUT Od THE rAsES. IT gOULv xApE wOU A xAnIrIAN, NO LESS THAN qAddAy! TO nO THyOUnH, SzEAp THE zASSgOyv:

Now ‘ALgAwS’ and ‘THyOUnH’ should correspond to ‘ALWAYS’ and ‘THROUGH’ respectively. Substituting letters we got till now gives

aREApER Od THIS rOvE WILL aE aLESSEv aY THE SlUEApY SzIRIT RESIvING IN THE HOLE. GO AHEAv, ANv dINv AWAY Od aREApING THE SzELL ON HIx rAST aY THE EsIL qAddAR. THE SzIRIT Od THE rAsE xAN IS ALWAYS WITH YOU. dINv THE xAGIr WANv THAT WILL LET YOU OUT Od THE rAsES. IT WOULv xApE YOU A xAGIrIAN, NO LESS THAN qAddAR! TO GO THROUGH, SzEAp THE zASSWORv:

Now “zASSWORv’, ‘aY’ and ‘EsIL’ should correspond to ‘PASSWORD’ , ‘BY’ and ‘EVIL’ respectively. Substituting we get.

BREApER Od THIS rODE WILL BE BLESSED BY THE SlUEApY SPIRIT RESIDING IN THE HOLE. GO AHEAD, AND dIND AWAY Od BREApING THE SPELL ON HIx rAST BY THE EVIL qAddAR. THE SPIRIT Od THE rAVE xAN IS ALWAYS WITH YOU. dIND THE xAGIr WAND THAT WILL LET YOU OUT Od THE rAVES. IT WOULD xApE YOU A xAGIrIAN, NO LESS THAN qAddAR! TO GO THROUGH, SPEAp THE PASSWORD:

Now ‘BREApER’ , ‘Od’, ‘HIx’ and ‘rAVES’ should correspond to ‘BREAKER’, ‘OF’, ‘HIM’ and ‘CAVES’ respectively. Substituting we get

BREAKER OF THIS CODE WILL BE BLESSED BY THE SlUEAKY SPIRIT RESIDING IN THE HOLE. GO AHEAD, AND FIND AWAY OF BREAKING THE SPELL ON HIM CAST BY THE EVIL qAFFAR. THE SPIRIT OF THE CAVE MAN IS ALWAYS WITH YOU. FIND THE MAGIC WAND THAT WILL LET YOU OUT OF THE CAVES. IT WOULD MAKE YOU A MAGICIAN, NO LESS THAN qAFFAR! TO GO THROUGH, SPEAK THE PASSWORD:

Finally ‘SlUEAKY’ should correspond to ‘SQUEAKY’ and the possibility of ‘q’ are ‘j’, ‘x’ and ‘z’. Substituting and Permuting possible password are OSE_NJQRM_DS, OSE_NZQRM_DS and OSE_NXQRM_DS of which OSE_NJQRM_DS was correct and entering the password we entered chapter 4.
The mapping of letters and original text has been provided below.

After all these steps we found out the following matches:

t	e	u	b	c	k	i	o	h	m	g	w	y	n	v	z	s	a	p	d	r	x	l
A	E	T	H	O	I	S	N	L	U	W	Y	R	G	D	P	V	B	K	F	C	M	Q

Table 1: Mapping of the words in the encrypted text

BREAKER OF THIS CODE WILL BE BLESSED BY THE SQUEAKY SPIRIT RESIDING IN THE HOLE. GO AHEAD, AND FIND AWAY OF BREAKING THE SPELL ON HIM CAST BY THE EVIL qAFFAR. THE SPIRIT OF THE CAVE MAN IS ALWAYS WITH YOU. FIND THE; MAGIC WAND THAT WILL LET YOU OUT OF THE CAVES. IT WOULD MAKE YOU A MAGICIAN, NO LESS THAN qAFFAR! TO GO THROUGH, SPEAK THE PASSWORD:

OSE_NJQRM_DS

Explanation of codes:-

1. The file input.txt contains the message to be deciphered.
2. The file preprocess.py removes all the characters except alphabets from the input.txt and it also changes the upper case letters into lower case letters and saves the final text into the file level_3.txt.
3. The file Frequency_Analysis.py creates the frequency histogram for the text to be deciphered.
4. The file Index_of_Coincidence.py calculates the index of the coincidence of the text.
5. The file permutation.py generates all the permutations of desired length and check on which permutation we get the maximum number of repetitions of words and corresponding to that permutation generate the plaintext in the file output.txt with the same structure as the file input.txt.
6. The file substitution.py substituted the letter to corresponding deciphered letters as we found them.

Reference:-

1. *https://en.wikipedia.org/wiki/Letter_frequency*
2. *https://en.wikipedia.org/wiki/Index_of_coincidence*