COS 738 Assignment 1 Task 2

Isa Tippens 4034973

Introduction

We are given the following encrypted text with the goal to decipher it and reveal the original message.

```
GFS WMY OG LGDVS MF SFNKYHOSU ESLLMRS, PC WS
BFGW POL DMFRQMRS, PL OG CPFU M UPCCSKSFO
HDMPFOSXO GC OIS LMES DMFRQMRS DGFR SFGQRI OG
CPDD GFS LISSO GK LG, MFU OISF WS NGQFO OIS
GNNQKKSFNSL GC SMNI DSOOSK. WS NMDD OIS EGLO
CKSJQSFODY GNNQKKPFR DSOOSK OIS 'CPKLO', OIS
FSXO EGLO GNNQKKPFR DSOOSK OIS 'LSNGFU' OIS
CGDDGWPFR EGLO GNNQKKPFR DSOOSK OIS 'OIPKU', MFU
LG GF, QFOPD WS MNNGQFO CGK MDD OIS UPCCSKSFO
DSOOSKL PF OIS HDMPFOSXO LMEHDS. OISF WS DGGB MO
OIS NPHISK OSXO WS WMFO OG LGDVS MFU WS MDLG
NDMLLPCY POL LYEAGDL. WS CPFU OIS EGLO GNNQKKPFR
LYEAGD MFU NIMFRS PO OG OIS CGKE GC OIS 'CPKLO'
DSOOSK GC OIS HDMPFOSXO LMEHDS, OIS FSXO EGLO
NGEEGF LYEAGD PL NIMFRSU OG OIS CGKE GC OIS
'LSNGFU' DSOOSK, MFU OIS CGDDGWPFR EGLO NGEEGF
LYEAGD PL NIMFRSU OG OIS CGKE GC OIS 'OIPKU'
DSOOSK, MFU LG GF, QFOPD WS MNNGQFO CGK MDD
LYEAGDL GC OIS NKYHOGRKME WS WMFO OG LGDVS..
```

There is no key given therefore ruling out key based ciphers or encryption. Shifting letters to N number of letters did not yield any results which left only substitution based deciphering. This means that each letter within the original text is mapped to another letter, the goal of this task would be to figure out the mapping and reverse the cipher.

Frequency Analysis

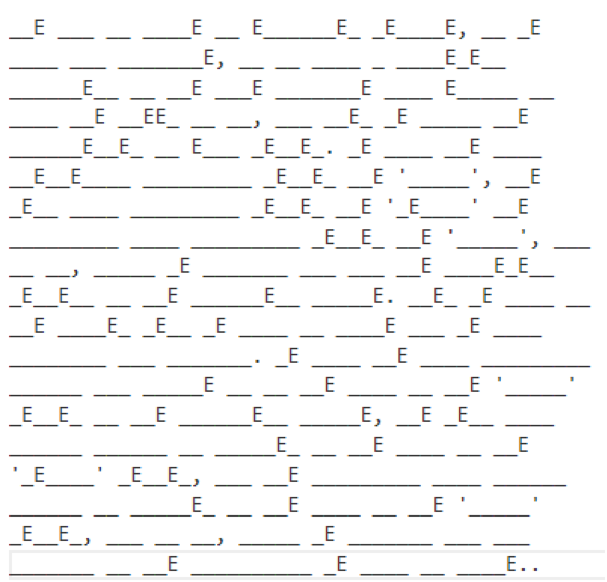
The first step to approaching this cipher is to count the occurrences of each letter and word within the ciphertext. Based on usage of letters and words within other texts, we can make assumptions of what the letters and words within the ciphertext should represent. The most common letters used within English are E, A, R, I, O, T, N, S , L and C from https://www.rd.com/article/common-letters-english-language/ and most common words include: the, of, and, a, to, in, is, you, that, it from https://www.espressoenglish.net/the-100-most-common-words-in-english/.

Using a python script, the following letters and words were counted:

S : 88	K : 35	U : 17	A : 5
O : 85	I : 33	R : 17	V : 3
G : 67	P : 30	W : 16	B : 2
F : 51	N : 29	Q : 14	J : 1
D : 42	C : 26	Y : 10	
L : 39	E : 23	H : 8	
M : 35	U : 17	X : 6	

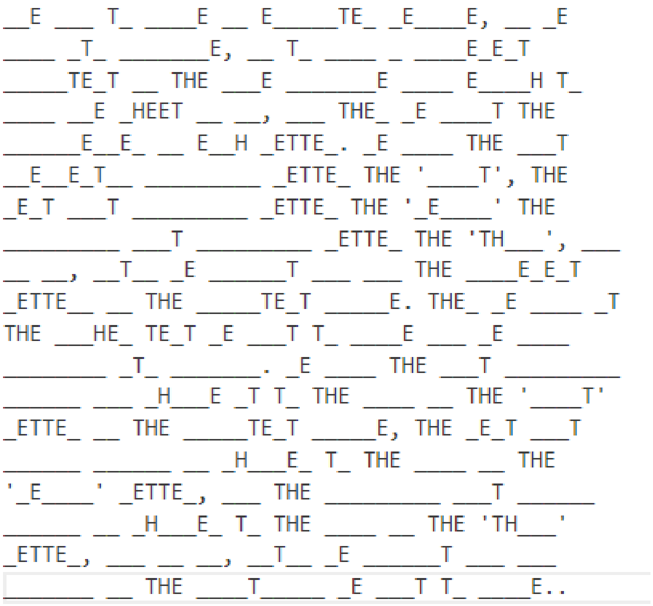
M : 1 MF : 1 PC : 1 GK : 1 PF : 1 MO : 1 PO : 1 GF : 2 PL : 3 LG : 3 GC : 7 OG : 8 WS : 10	WMY : 1 GFS : 2 POL : 2 CGK : 2 MDD : 2 MFU : 6 OIS : 22 BFGW : 1 LMES : 1 DGFR : 1 CPDD : 1 SMNI : 1 NMDD : 1 DGGB : 1 OSXO : 1 MDLG : 1 CPFU : 2 OISF : 2 FSXO : 2 WMFO : 2 CGKE : 3 EGLO : 6
LISSO : 1 NGQFO : 1 CPKLO : 2 OIPKU : 2 QFOPD : 2 LGDVS : 3 SFGQRI : 1 NPHISK : 1 NIMFRS : 1 LSNGFU : 2 LMEHDS : 2 NGEEGF : 2 LYEAGD : 3 DSOOSK : 7	ESLLMRS : 1 DSOOSKL : 1 MNNGQFO : 2 LYEAGDL : 2 NIMFRSU : 2 NDMLLPCY : 1 DMFRQMRS : 2 SFNKYHOSU : 1 UPCCSKSFO : 2 CGDDGWPFR : 2 HDMPFOSXO : 3 GNNQKKPFR : 4 CKSJQSFODY : 1 NKYHOGRKME : 1 GNNQKKSFNSL : 1

As shown in the first table, the most common letter is S which we will associate with E. We can replace all S letters in the ciphertext with E and leave unconverted letters as _.



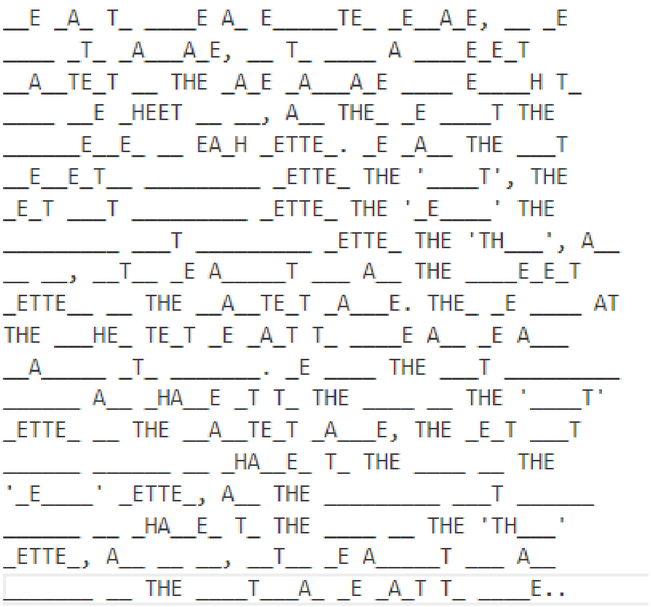
```
__E ___ __ ____E __ E______E_ _E____E, __ _E
____ ___ _______E, __ __ ____ _ ____E_E__
______E__ __ __E ___E _______E ____ E_____ __
____ __E __EE_ __ __, ___ __E_ _E _____ __E
______E__E_ __ E___ _E__E_. _E ____ __E ____
__E__E____ _________ _E__E_ __E '_____', __E
_E__ ____ _________ _E__E_ __E '_E____' __E
_________ ____ _________ _E__E_ __E '_____', ___
__ __, _____ _E _______ ___ ___ __E ____E_E__
_E__E__ __ __E ______E__ _____E. __E_ _E ____ __
__E ____E_ _E__ _E ____ __ ____E ___ _E ____
________ ___ _______. _E ____ __E ____ _________
______ ___ _____E __ __ __E ____ __ __E '_____'
_E__E_ __ __E ______E__ _____E, __E _E__ ____
______ ______ __ _____E_ __ __E ____ __ __E
'_E____' _E__E_, ___ __E _________ ____ ______
______ __ _____E_ __ __E ____ __ __E '_____'
_E__E_, ___ __ __, _____ _E _______ ___ ___
_______ __ __E __________ _E ____ __ ____E..
```

The very first word within the ciphertext, GFS, ends with an S which could be mean that the original word is THE, however, the most common word within the ciphertext is OIS which also ends with an S therefore O and I will be replaced with T and H respectively.



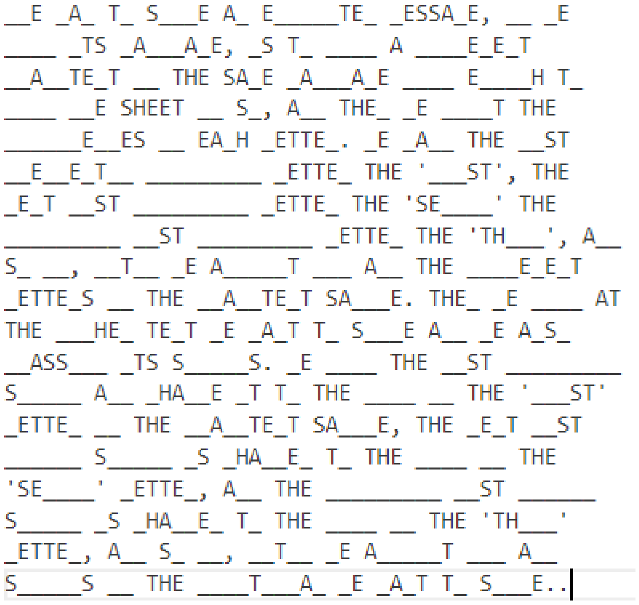
```
__E ___ T_ ____E __ E_____TE_ _E____E, __ _E
____ _T_ _______E, __ T_ ____ _ ____E_E_T
_____TE_T __ THE ___E _______E ____ E____H T_
____ __E _HEET __ __, ___ THE_ _E ____T THE
______E__E_ __ E__H _ETTE_. _E ____ THE ___T
__E__E_T__ _________ _ETTE_ THE '____T', THE
_E_T ___T _________ _ETTE_ THE '_E____' THE
_________ ___T _________ _ETTE_ THE 'TH___', ___
__ __, __T__ _E ______T ___ ___ THE ____E_E_T
_ETTE__ __ THE _____TE_T _____E. THE_ _E ____ _T
THE ___HE_ TE_T _E ___T T_ ____E ___ _E ____
________ _T_ _______. _E ____ THE ___T _________
______ ___ _H___E _T T_ THE ____ __ THE '____T'
_ETTE_ __ THE _____TE_T _____E, THE _E_T ___T
______ ______ __ _H___E_ T_ THE ____ __ THE
'_E____' _ETTE_, ___ THE _________ ___T ______
______ __ _H___E_ T_ THE ____ __ THE 'TH___'
_ETTE_, ___ __ __, __T__ _E ______T ___ ___
_______ __ THE ____T_____ _E ___T T_ ____E..
```

Another letter of note is M where there is a case that it is alone within the ciphertext, this could be either mapped from A or I. From the article, A is the most common letter therefore M will be mapped to A



```
__E _A_ T_ ____E A_ E_____TE_ _E__A_E, __ _E
____ _T_ _A___A_E, __ T_ ____ A ____E_E_T
__A__TE_T __ THE _A_E _A___A_E ____ E____H T_
____ __E _HEET __ __, A__ THE_ _E ____T THE
______E__E_ __ EA_H _ETTE_. _E _A__ THE ___T
__E__E_T__ _________ _ETTE_ THE '____T', THE
_E_T ___T _________ _ETTE_ THE '_E____' THE
_________ ___T _________ _ETTE_ THE 'TH___', A__
__ __, __T__ _E A_____T ___ A__ THE ____E_E_T
_ETTE__ __ THE __A__TE_T _A___E. THE_ _E ____ AT
THE ___HE_ TE_T _E _A_T T_ ____E A__ _E A___
__A_____ _T_ _______. _E ____ THE ___T _________
______ A__ _HA__E _T T_ THE ____ __ THE '____T'
_ETTE_ __ THE __A__TE_T _A___E, THE _E_T ___T
______ ______ __ _HA__E_ T_ THE ____ __ THE
'_E____' _ETTE_, A__ THE _________ ___T ______
______ __ _HA__E_ T_ THE ____ __ THE 'TH___'
_ETTE_, A__ __ __, __T__ _E A_____T ___ A__
_______ __ THE ____T___A_ _E _A_T T_ ____E..
```

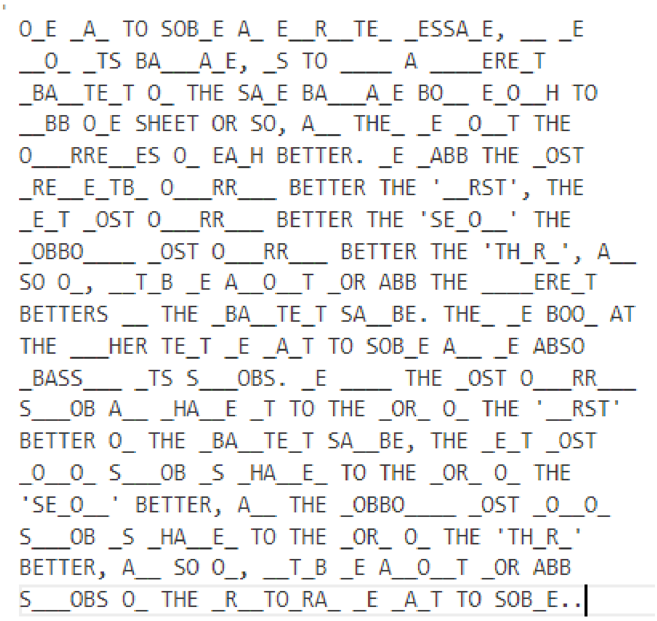
There is a number of words in our decrypted text that contains _ETTE_ which is DSOOSK within the ciphertext, notably there is an occurrence of DSOOSKL which could mean that its plural, therefore L will be mapped to S. This also reveals the word SHEET in the decrypted text.



```
__E _A_ T_ S___E A_ E_____TE_ _ESSA_E, __ _E
____ _TS _A___A_E, _S T_ ____ A ____E_E_T
__A__TE_T __ THE SA_E _A___A_E ____ E____H T_
____ __E SHEET __ S_, A__ THE_ _E ____T THE
______E__ES __ EA_H _ETTE_. _E _A__ THE __ST
__E__E_T__ _________ _ETTE_ THE '___ST', THE
_E_T __ST _________ _ETTE_ THE 'SE____' THE
_________ __ST _________ _ETTE_ THE 'TH___', A__
S_ __, __T__ _E A_____T ___ A__ THE ____E_E_T
_ETTE_S __ THE __A__TE_T SA___E. THE_ _E ____ AT
THE ___HE_ TE_T _E _A_T T_ S___E A__ _E A_S_
__ASS___ _TS S_____S. _E ____ THE __ST _________
S_____ A__ _HA__E _T T_ THE ____ __ THE '___ST'
_ETTE_ __ THE __A__TE_T SA___E, THE _E_T __ST
______ S_____ _S _HA__E_ T_ THE ____ __ THE
'SE____' _ETTE_, A__ THE _________ __ST ______
S_____ _S _HA__E_ T_ THE ____ __ THE 'TH___'
_ETTE_, A__ S_ __, __T__ _E A_____T ___ A__
S_____S __ THE ____T___A_ _E _A_T T_ S___E..
```

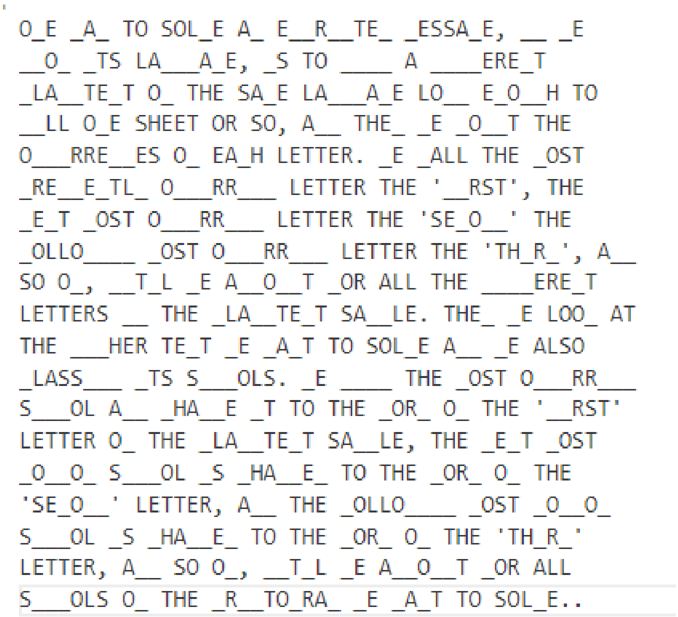
There is 8 occurrences of OG which currently converts to T_, the only word we can be formed from this is TO. Returning to DSOOSK (_ETTE_), there is two possibilities being BETTER or LETTER

DSOOSK => BETTER



```
O_E _A_ TO SOB_E A_ E__R__TE_ _ESSA_E, __ _E
__O_ _TS BA___A_E, _S TO ____ A ____ERE_T
_BA__TE_T O_ THE SA_E BA___A_E BO__ E_O__H TO
__BB O_E SHEET OR SO, A__ THE_ _E _O__T THE
O___RRE__ES O_ EA_H BETTER. _E _ABB THE _OST
_RE__E_TB_ O___RR___ BETTER THE '__RST', THE
_E_T _OST O___RR___ BETTER THE 'SE_O__' THE
_OBBO____ _OST O___RR___ BETTER THE 'TH_R_', A__
SO O_, __T_B _E A__O__T _OR ABB THE ____ERE_T
BETTERS __ THE _BA__TE_T SA__BE. THE_ _E BOO_ AT
THE ___HER TE_T _E _A_T TO SOB_E A__ _E ABSO
_BASS___ _TS S___OBS. _E ____ THE _OST O___RR___
S___OB A__ _HA__E _T TO THE _OR_ O_ THE '__RST'
BETTER O_ THE _BA__TE_T SA__BE, THE _E_T _OST
_O__O_ S___OB _S _HA__E_ TO THE _OR_ O_ THE
'SE_O__' BETTER, A__ THE _OBBO____ _OST _O__O_
S___OB _S _HA__E_ TO THE _OR_ O_ THE 'TH_R_'
BETTER, A__ SO O_, __T_B _E A__O__T _OR ABB
S___OBS O_ THE _R__TO_RA_ _E _A_T TO SOB_E..
```

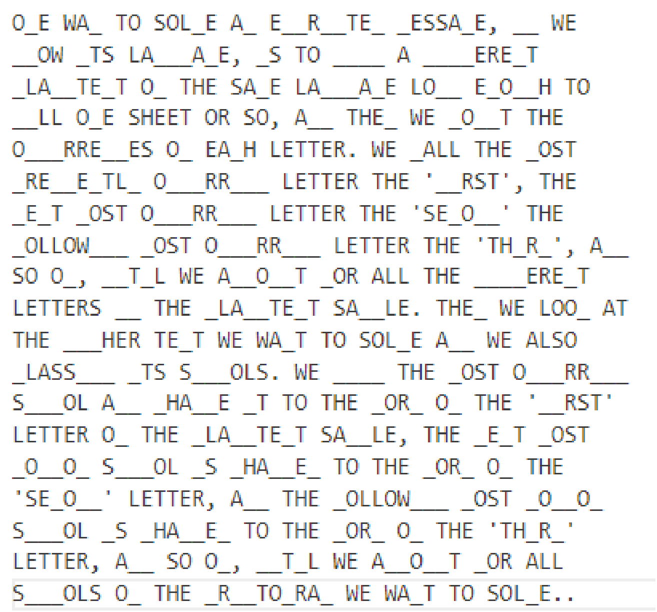
DSOOSK => LETTER



```
O_E _A_ TO SOL_E A_ E__R__TE_ _ESSA_E, __ _E
__O_ _TS LA___A_E, _S TO ____ A ____ERE_T
_LA__TE_T O_ THE SA_E LA___A_E LO__ E_O__H TO
__LL O_E SHEET OR SO, A__ THE_ _E _O__T THE
O___RRE__ES O_ EA_H LETTER. _E _ALL THE _OST
_RE__E_TL_ O___RR___ LETTER THE '__RST', THE
_E_T _OST O___RR___ LETTER THE 'SE_O__' THE
_OLLO____ _OST O___RR___ LETTER THE 'TH_R_', A__
SO O_, __T_L _E A__O__T _OR ALL THE ____ERE_T
LETTERS __ THE _LA__TE_T SA__LE. THE_ _E LOO_ AT
THE ___HER TE_T _E _A_T TO SOL_E A__ _E ALSO
_LASS___ _TS S___OLS. _E ____ THE _OST O___RR___
S___OL A__ _HA__E _T TO THE _OR_ O_ THE '__RST'
LETTER O_ THE _LA__TE_T SA__LE, THE _E_T _OST
_O__O_ S___OL _S _HA__E_ TO THE _OR_ O_ THE
'SE_O__' LETTER, A__ THE _OLLO____ _OST _O__O_
S___OL _S _HA__E_ TO THE _OR_ O_ THE 'TH_R_'
LETTER, A__ SO O_, __T_L _E A__O__T _OR ALL
S___OLS O_ THE _R__TO_RA_ _E _A_T TO SOL_E..
```

LETTER is the better option as for DSOOSKL it becomes LETTERS and also reveals the word ALL within the text.

Looking at WS (_E), there is 4 possibilities: BE, ME, WE, HE. Using W => W, the decrypted text starts to become more clear and we can start filling in blank areas.



```
O_E WA_ TO SOL_E A_ E__R__TE_ _ESSA_E, __ WE
__OW _TS LA___A_E, _S TO ____ A ____ERE_T
_LA__TE_T O_ THE SA_E LA___A_E LO__ E_O__H TO
__LL O_E SHEET OR SO, A__ THE_ WE _O__T THE
O___RRE__ES O_ EA_H LETTER. WE _ALL THE _OST
_RE__E_TL_ O___RR___ LETTER THE '__RST', THE
_E_T _OST O___RR___ LETTER THE 'SE_O__' THE
_OLLOW___ _OST O___RR___ LETTER THE 'TH_R_', A__
SO O_, __T_L WE A__O__T _OR ALL THE ____ERE_T
LETTERS __ THE _LA__TE_T SA__LE. THE_ WE LOO_ AT
THE ___HER TE_T WE WA_T TO SOL_E A__ WE ALSO
_LASS___ _TS S___OLS. WE ____ THE _OST O___RR___
S___OL A__ _HA__E _T TO THE _OR_ O_ THE '__RST'
LETTER O_ THE _LA__TE_T SA__LE, THE _E_T _OST
_O__O_ S___OL _S _HA__E_ TO THE _OR_ O_ THE
'SE_O__' LETTER, A__ THE _OLLOW___ _OST _O__O_
S___OL _S _HA__E_ TO THE _OR_ O_ THE 'TH_R_'
LETTER, A__ SO O_, __T_L WE A__O__T _OR ALL
S___OLS O_ THE _R__TO_RA_ WE WA_T TO SOL_E..
```

Filling in the blanks

With the remaining words we can guess the actual word based on context from either surrounding words or words from the existing letters. Each mapping reveals more of other words and sometimes solves words.

Encrypted text (current decrypted text) = Guess

EGLO (_OST) = MOST, POST, HOST, COST

NMDD (_ALL) = BALL, MALL, WALL, TALL, GALL, YALL, FALL, CALL

CGKE (_ORM) = FORM, DORM, WORM

GNNQKKSFNSL => OCC_RRE_CES = OCCURRENCES

ESLLMRS (MESSA_E) = MESSAGE

UPCCSKSFO (__FFERENT) = DIFFERENT

SFNKYHOSU (ENCR__TED) = ENCRYPTED

HDMPFOSXO (PLAINTE_T) = PLAINTEXT

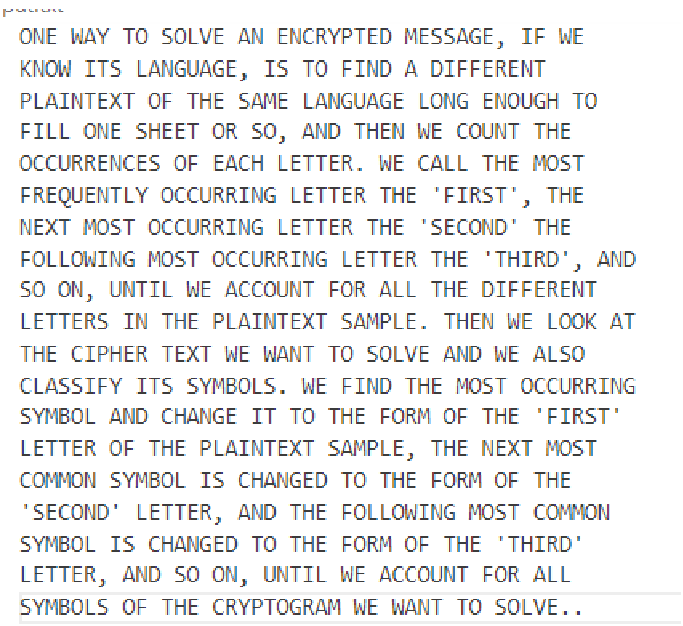
LGDVS (SOL_E) = SOLVE

BFGW (_NOW) = KNOW

LYEAGD (SYM_OL) = SYMBOL

CKSJQSFODY (FRE_UENTLY) = FREQUENTLY

After all letters are converted, we are left with the deciphered text:



```
ONE WAY TO SOLVE AN ENCRYPTED MESSAGE, IF WE
KNOW ITS LANGUAGE, IS TO FIND A DIFFERENT
PLAINTEXT OF THE SAME LANGUAGE LONG ENOUGH TO
FILL ONE SHEET OR SO, AND THEN WE COUNT THE
OCCURRENCES OF EACH LETTER. WE CALL THE MOST
FREQUENTLY OCCURRING LETTER THE 'FIRST', THE
NEXT MOST OCCURRING LETTER THE 'SECOND' THE
FOLLOWING MOST OCCURRING LETTER THE 'THIRD', AND
SO ON, UNTIL WE ACCOUNT FOR ALL THE DIFFERENT
LETTERS IN THE PLAINTEXT SAMPLE. THEN WE LOOK AT
THE CIPHER TEXT WE WANT TO SOLVE AND WE ALSO
CLASSIFY ITS SYMBOLS. WE FIND THE MOST OCCURRING
SYMBOL AND CHANGE IT TO THE FORM OF THE 'FIRST'
LETTER OF THE PLAINTEXT SAMPLE, THE NEXT MOST
COMMON SYMBOL IS CHANGED TO THE FORM OF THE
'SECOND' LETTER, AND THE FOLLOWING MOST COMMON
SYMBOL IS CHANGED TO THE FORM OF THE 'THIRD'
LETTER, AND SO ON, UNTIL WE ACCOUNT FOR ALL
SYMBOLS OF THE CRYPTOGRAM WE WANT TO SOLVE..
```

App Repository

https://github.com/IsaTippens/DecipherApp

Video Demo

https://drive.google.com/file/d/1HsdFpo3TTfOOkBJa7JNntxUpnMDa39uj/view?usp=drive_link