Classical Ciphers

Cryptography

School of Engineering and Technology
CQUniversity Australia

Prepared by Steven Gordon on 04 Jan 2022,
classical.tex, r1964

Contents

Caesar Cipher (algorithm)

To encrypt with a key `k`, shift each letter of the plaintext `k` positions to the right in the alphabet, wrapping back to the start of the alphabet if necessary. To decrypt, shift each letter of the ciphertext `k` positions to the left (wrapping if necessary).

Caesar Cipher Encryption (exercise)

Using the Caesar cipher, encrypt plaintext `hello` with key 3.

How many keys are possible in the Caesar cipher? (question)

If the Caesar cipher is operating on the characters a–z, then how many possible keys are there? Is a key of 0 possible? Is it a good choice? What about a key of 26?

Caesar Cipher Decryption (exercise)

You have received the ciphertext `TBBQOLR`. You know the Caesar cipher was used with key `n`. Find the plaintext.

Caesar Cipher, formal (algorithm)

$$C = E(K, P) = (P + K) \bmod 26 \quad (1)$$

$$P = D(K, C) = (C - K) \bmod 26 \quad (2)$$

Caesar Cipher, formal (exercise)

Consider the following mapping.

a	b	c	d	e	f	g	h	i	j	k	l	m
0	1	2	3	4	5	6	7	8	9	10	11	12
n	o	p	q	r	s	t	u	v	w	x	y	z
13	14	15	16	17	18	19	20	21	22	23	24	25

Use the the formal (mathematical) algorithm for Caesar cipher to decrypt `SDV` with key `p`.

Caesar Encrypt and Decrypt (python)

```
>>> pycipher.Caesar(3).encipher("hello")
'KHOOR'
>>> pycipher.Caesar(3).decipher("khoor")
'HELLO'
```

Brute Force Attack (definition)

Try all combinations (of keys) until the correct plaintext/key is found.

Caesar Brute Force (exercise)

The ciphertext `FRUURJVBCANNC` was obtained using the Caesar cipher. Find the plaintext using a brute force attack.

Caesar Brute Force (python)

```
for k in range(0,26):
pycipher.Caesar(k).decipher("FRUURJVBCANNC")
```

Caesar Brute Force Results (text)

```
0: FRUURJVBCANNC 13: SEHHEWIOPNAAP
1: EQTTQIUABZMMB 14: RDGGDVHNOMZZO
2: DPSSPHTZAYLLA 15: QCFFCUGMNLYYN
3: CORROGSYZXKKZ 16: PBEEBTFLMKXXM
4: BNQQNFRXYWJJY 17: OADDASEKLJWWL
5: AMPPMEQWXVIIX 18: NZCCZRDJKIVVK
6: ZLOOLDPVWUHHW 19: MYBBYQCIJHUUJ
7: YKNNKCOUVTGGV 20: LXAAXPBHIGTTI
8: XJMMJBNTUSFFU 21: KWZZWOAGHFSSH
9: WILLIAMSTREET 22: JVYYVNZFGERRG
10: VHKKHZLRSQDDS 23: IUXXUMYEFDQQF
11: UGJJGYKQRPCCR 24: HTWWTLXDECPPE
12: TFIIFXJPQOBBQ 25: GSVVSKWCDBOOD
```

How many attempts for Caesar brute force? (question)

What is the worst, best and average case of number of attempts to brute force ciphertext obtained using the Caesar cipher?

Recognisable Plaintext upon Decryption (assumption)

The decrypter will be able to recognise that the plaintext is correct (and therefore the key is correct). Decrypting ciphertext using the incorrect key will *not* produce the original plaintext. The decrypter will be able to recognise that the key is wrong, i.e. the decryption will produce unrecognisable output.

Is plaintext always recognisable? (question)

Caesar cipher is using recognisably correct plaintext, i.e. English words. But is the correct plaintext always recognisable? What if the plaintext was a different language? Or compressed? Or it was an image or video? Or binary file, e.g. .exe? Or a set of characters chosen randomly, e.g. a key or password?

How to improve upon the Caesar cipher?

1. Increase the key space so brute force is harder
2. Change the plaintext (e.g. compress it) so harder to recognise structure

Contents

Permutation (definition)

A permutation of a finite set of elements is an ordered sequence of all the elements of S, with each element appearing exactly once. In general, there are $n!$ permutations of a set with n elements.

Permutation (example)

Consider the set $S = \{a, b, c\}$. There are six permutations of S:

`abc, acb, bac, bca, cab, cba`

This set has 3 elements. There are $3! = 3 \times 2 \times 1 = 6$ permutations.

Monoalphabetic (Substitution) Cipher (definition)

Given the set of possible plaintext letters (e.g. English alphabetc, a–z), a single permutation is chosen and used to determine the corresponding ciphertext letter.

Monoalphabetic (Substitution) Cipher (example)

In advance, the sender and receiver agree upon a permutation to use, e.g.:

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

To encrypt the plaintext `hello`, the agreed upon permutation (or mapping) is used to produce the ciphertext `ESCCF`.

Decrypt Monoalphabetic Cipher (exercise)

Decrypt the ciphertext `QSWBSR` using the permutation chosen in the previous example.

How many keys in English monoalphabetic cipher? (question)

How many possible keys are there for a monoalphabetic cipher that uses the English lowercase letters? What is the length of an actual key?

Brute Force on Monoalphabetic Cipher (exercise)

You have intercepted a ciphertext message that was obtained with an English monoalphabetic cipher. You have a Python function called:
`mono_decrypt_and_check(ciphertext,key)`
that decrypts the ciphertext with a key, and returns the plaintext if it is correct, otherwise returns false. You have tested the Python function in a while loop and the computer can apply the function at a rate of 1,000,000,000 times per second. Find the average time to perform a brute force on the ciphertext.

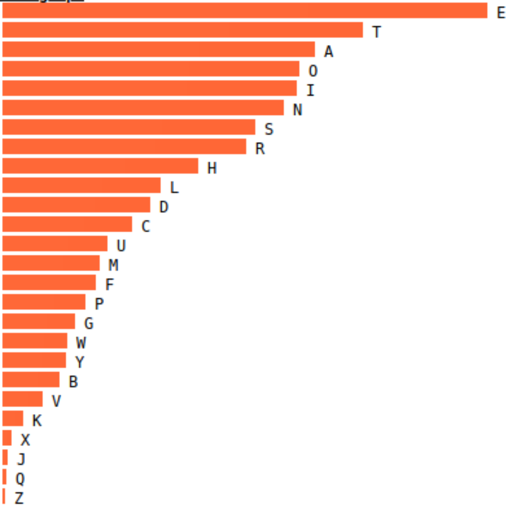
Frequency Analysis Attack (definition)

Find (portions of the) key and/or plaintext by using insights gained from comparing the actual frequency of letters in the ciphertext with the expected frequency of letters in the plaintext. Can be expanded to analyse sets of letters, e.g. digrams, trigrams, n-grams, words.

Relative Frequency of Letters by Norvig

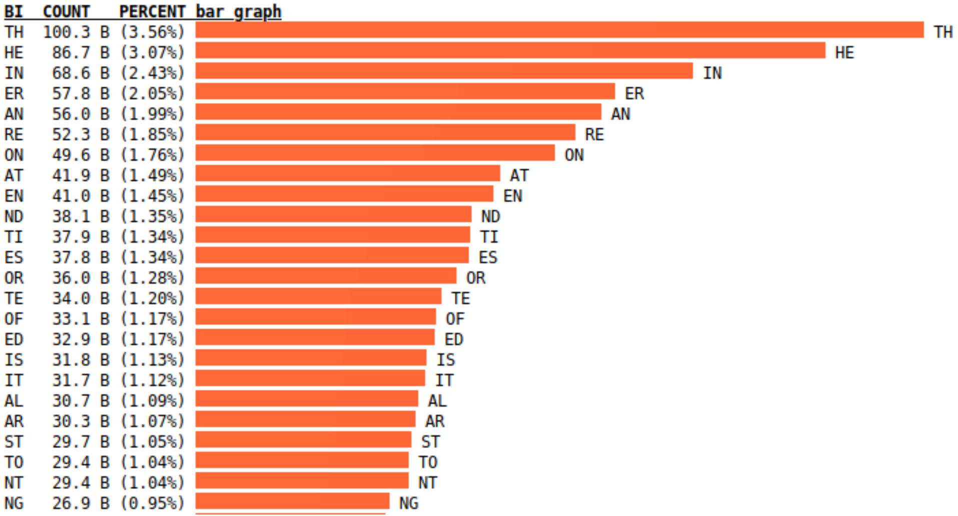
LET	COUNT	PERCENT
E	445.2 B	12.49%
T	330.5 B	9.28%
A	286.5 B	8.04%
O	272.3 B	7.64%
I	269.7 B	7.57%
N	257.8 B	7.23%
S	232.1 B	6.51%
R	223.8 B	6.28%
H	180.1 B	5.05%
L	145.0 B	4.07%
D	136.0 B	3.82%
C	119.2 B	3.34%
U	97.3 B	2.73%
M	89.5 B	2.51%
F	85.6 B	2.40%
P	76.1 B	2.14%
G	66.6 B	1.87%
W	59.7 B	1.68%
Y	59.3 B	1.66%
B	52.9 B	1.48%
V	37.5 B	1.05%
K	19.3 B	0.54%
X	8.4 B	0.23%
J	5.7 B	0.16%
Q	4.3 B	0.12%
Z	3.2 B	0.09%

bar graph



Credit: *Letter Counts* by Peter Norvig

Relative Frequency of Digrams by Norvig



Credit: *Two-Letter Sequence (Bigram) Counts* by Peter Norvig

Relative Frequency of N-Grams by Norvig

1	2grams	3grams	4-grams	5-grams	6-grams	7-grams	8-grams	9-grams
e	th	the	tion	ation	ations	present	differen	different
t	he	and	atio	tions	ration	ational	national	governmen
a	in	ing	that	which	tional	through	consider	overnment
o	er	ion	ther	ction	nation	between	position	formation
i	an	tio	with	other	ection	ication	ifferent	character
n	re	ent	ment	their	cation	differe	governme	velopment
s	on	ati	ions	there	lation	ifferen	vernment	developme
r	at	for	this	ition	though	general	overnmen	evelopmen
h	en	her	here	ement	presen	because	interest	condition
l	nd	ter	from	inter	tation	develop	importan	important
d	ti	hat	ould	ional	should	america	ormation	articular
c	es	tha	ting	ratio	resent	however	formatio	particula
u	or	ere	hich	would	genera	eration	relation	represent
m	te	ate	whic	tiona	dition	nationa	question	individua
f	of	his	ctio	these	ationa	conside	american	ndividual
p	ed	con	ence	state	produc	onsider	characte	relations
g	is	res	have	natio	throug	ference	haracter	political
w	it	ver	othe	thing	hrough	positio	articula	informati
y	al	all	ight	under	etween	osition	possible	nformatio
b	ar	ons	sion	ssion	betwee	ization	children	universit
v	st	nce	ever	ectio	differ	fferent	elopment	following
k	to	men	ical	catio	icatio	without	velopmen	experienc
x	nt	ith	they	latio	people	ernment	developm	stitution
j	ng	ted	inte	about	iffere	vernmen	evelopme	xperience
q	se	ers	ough	count	fferen	overnme	conditio	education
z	ha	pro	ance	ments	struct	governm	ondition	roduction

Credit: *N-Letter Sequences (N-grams)"* by Peter Norvig

Break a Monoalphabetic Cipher (exercise)

Ciphertext:

```
ziolegxkltqodlzgofzkgrxetngxzgzithkofeohs
tlqfrzteifojxtlgyltexkofuegdhxztklqfregd
hxztkftzvgkalvoziygexlgfofztkftzltexkoznz
itegxkltoltyytezoctsnlhsozofzgzvghqkzlyo
klzofzkgrxeofuzitzitgkngyeknhzgukqhinofes
xrofuigvdqfnesqlloeqsqfrhghxsqkqsugkozid
lvgkaturtlklqrouozqsloufqzxktlqfrltegfrhk
gcorofurtzqoslgyktqsofztkftzltexkoznhkgz
gegslqsugkozidlqfrziktqzltuohltecokxltlyo
ktvqsslitfetngxvossstqkfwgzizitgktzoeqsq
lhtezlgyegdhxztkqfrftzvgkaltexkoznqlvtssq
ligvziqzzitgknolqhhsotrofzitofztkftzziol
afgvstrutvossitshngxofrtloufofuqfrrtctsgh
ofultexktqhhsoeqzogflqfrftzvgkahkgzgegsl
qlvtssqlwxosrofultexktftzvgkal
```

Contents

Playfair Matrix Construction (algorithm)

Write the letters of keyword `k` row-by-row in a 5-by-5 matrix. Do not include duplicate letters. Fill the remainder of the matrix with the alphabet. Treat the letters *i* and *j* as the same (that is, they are combined in the same cell of the matrix).

Playfair Matrix Construction (exercise)

Construct the Playfair matrix using keyword `australia`.

Playfair Encryption (algorithm)

Split the plaintext into pairs of letters. If a pair has identical letters, then insert a special letter x in between. If the resulting set of letters is odd, then pad with a special letter x.

Locate the plaintext pair in the Playfair matrix. If the pair is on the same column, then shift each letter down one cell to obtain the resulting ciphertext pair. Wrap when necessary. If the plaintext pair is on the same row, then shift to the right one cell. Otherwise, the first ciphertext letter is that on the same row as the first plaintext letter and same column as the second plaintext letter, and the second ciphertext letter is that on the same row as the second plaintext letter and same column as the first plaintext letter.

Repeat for all plaintext pairs.

Playfair Encryption (exercise)

Find the ciphertext if the Playfair cipher is used with keyword `australia` and plaintext `hello`.

Does Playfair cipher always map a letter to the same ciphertext letter? (question)

Using the Playfair cipher with keyword `australia`, encrypt the plaintext `hellolove`.

With the Playfair cipher, if a letter occurs multiple times in the plaintext, will that letter always encrypt to the same ciphertext letter?

If a pair of letters occurs multiple times, will that pair always encrypt to the same ciphertext pair?

Is the Playfair cipher subject to frequency analysis attacks?

Contents

Polyalphabetic (Substitution) Cipher (definition)

Use a different monoalphabetic substitution as proceeding through the plaintext. A key determines which monoalphabetic substitution is used for each transformation.

Examples of Polyalphabetic Ciphers

- Vigenère Cipher: uses Caesar cipher, but Caesar key changes each letter based on keyword
- Vernam Cipher: binary version of Vigenère, using XOR
- One Time Pad: same as Vigenère/Vernam, but random key as long as plaintext

Contents

Vigenère Cipher (algorithm)

For each letter of plaintext, a Caesar cipher is used. The key for the Caesar cipher is taken from the Vigenère key(word), progressing for each letter and wrapping back to the first letter when necessary. Formally, encryption using a keyword of length m is:

$$c_i = (p_i + k_{i \bmod m}) \bmod 26$$

where p_i is letter i (starting at 0) of plaintext P, and so on.

Vigenère Cipher Encryption (example)

Using the Vigenère cipher to encrypt the plaintext `carparkbehindsupermarket` with the keyword `sydney` produces the ciphertext `UYUCEPCZHUMLVQXCIPEYUXIR`. The keyword would be repeated when Caesar is applied:

```
P: carparkbehindsupermarket
K: sydneysydneysydneysydney
C: UYUCEPCZHUMLVQXCIPEYUXIR
```

Vigenère Cipher Encryption (exercise)

Use Python (or other software tools) to encrypt the plaintext `centralqueensland` with the following keys with the Vigenère cipher, and investigate any possible patterns in the ciphertext: `cat`, `dog`, `a`, `giraffe`.

Weakness of Vigenère Cipher

- Determine the length of the keyword m
  - Repeated n-grams in the ciphertext may indicate repeated n-grams in the plaintext
  - Separation between repeated n-grams indicates possible keyword length m
  - If plaintext is long enough, multiple repetitions make it easier to find m
- Treat the ciphertext as that from m different monoalphabetic ciphers
  - E.g. Caesar cipher with m different keys
  - Break the monoalphabetic ciphers with frequency analysis
- With long plaintext, and repeating keyword, Vigenère can be broken

Breaking Vigenère Cipher (example)

Ciphertext `ZICVTWQNGRZGVTWAVZHCQYGLMGJ` has repetition of `VTW`. That suggests repetition in the plaintext at the same position, which would be true if the keyword repeated at the same position.

```
012345678901234567890123456
ZICVTWQNGRZGVTWAVZHCQYGLMGJ
```

That is, it is possible the key letter at position 3 is the repated at position 12. That in turn suggest a keyword length of 9 or 3.

```
ciphertext ZICVTWQNGRZGVTWAVZHCQYGLMGJ
length=3:  012012012012012012012012012
length=9:  012345678012345678012345678
```

An attacker would try both keyword lengths. With a keyword length of 9, the attacker then performs Caesar cipher frequency analysis on every 9th letter. Eventually they find plaintext is `wearediscoveredsaveyourself` and keyword is `deceptive`.

Contents

Vernam Cipher (algorithm)

Encryption is performed as:

$$c_i = p_i \oplus k_i$$

decryption is performed as:

$$p_i = c_i \oplus k_i$$

where p_i is the ith bit of plaintext, and so on. The key is repeated where necessary.

XOR (python)

```
>>> def xor(x, y):
... return '{1:0{0}b}'.format(len(x), int(x, 2) ^ int(y, 2))
...
```

Vernam Cipher Encryption (exercise)

Using the Vernam cipher, encrypt the plaintext 0111010101010000011011001 with the key 01011.

Vernam Cipher Encryption (python)

```
>>> xor('0111010101010000110110 01','010110101101011010110101')
'0010111110000110011011 00'
```

Caesar Cipher
Monoalphabetic Ciphers
Playfair Cipher
Polyalphabetic Ciphers
Vigenère Cipher
Vernam Cipher
One Time Pad
Transposition Techniques

Contents

One-Time Pad (algorithm)

Use polyalphabetic cipher (such as Vigenère or Vernam) but where the key must be: random, the same length as the plaintext, and not used multiple times.

Properties of OTP

- Encrypting plaintext with random key means output ciphertext will be random
  - E.g. XOR plaintext with a random key produces random sequence of bits in ciphertext
- Random ciphertext contains no information about the structure of plaintext
  - Attacker cannot analyse ciphertext to determine plaintext
- Brute force attack on key is ineffective
  - Multiple different keys will produce recognisable plaintext
  - Attacker has no way to determine which of the plaintexts are correct
- OTP is only known unbreakable (unconditionally secure) cipher

Attacking OTP (example)

Consider a variant of Vigenère cipher that has 27 characters (including a space). An attacker has obtained the ciphertext:

`ANKYODKYUREPFJBYOJDSPLREYIUNOFDOIUERFPLUYTS`

Attacker tries all possible keys. Two examples:

```
k1:  pxlmvmsydofuyrvzwc tnlebnecvgdupahfzzlmnyih
p1:  mr mustard with the candlestick in the hall
k2:  pftgpmiydgaxgoufhklllmhsqdqogtewbqfgyovuhwt
p2:  miss scarlet with the knife in the library
```

There are many other legible plaintexts obtained with other keys. No way for attacker to know the correct plaintext

Summary of OTP

- Only known unbreakable (unconditionally secure) cipher
  - Ciphertext has no statistical relationship with plaintext
  - Given two potential plaintext messages, attacker cannot identify the correct message
- But two significant practical limitations:
  1. Difficult to create large number of random keys
  2. Distributing unique long random keys is difficult
- Limited practical use

Contents

Transposition vs Substitution

- Substitution: replace one (or more) character in plaintext with another from the entire possible character set
- Transposition: re-arrange the characters in the plaintext
  - The set of characters in the ciphertext is the same as in the plaintext
  - Problem: the plaintext frequency statistics are also in the ciphertext
- On their own, transposition techniques are easy to break
- Combining transposition with substitution makes ciphers stronger, and building block of modern ciphers

Rail Fence Cipher Encryption (definition)

Select a depth as a key. Write the plaintext in diagonals in a zig-zag manner to the selected depth. Read row-by-row to obtain the ciphertext.

Rail Fence Encryption (exercise)

Consider the plaintext `securityandcryptography` with key 4. Using the rail fence cipher, find the ciphertext.

Rows Columns Cipher Encryption (definition)

Select a number of columns m and permutate the integers from 1 to m to be the key. Write the plaintext row-by-row over m columns. Read column-by-column, in order of the columns determined by the key, to obtain the ciphertext.

Rows Columns Encryption (exercise)

Consider the plaintext `securityandcryptography` with key 315624. Using the rows columns cipher, find the ciphertext.

Rows Columns Multiple Encryption (example)

Assume the ciphertext from the previous example has been encrypted again with the same key. The resulting ciphertext is `YYCPRRCTEOIPDRAHYSGUATXH`. Now let's view how the cipher has "mixed up" the letters of the plaintext. If the plaintext letters are numbered by position from 01 to 24, their order (split across two rows) is:

01 02 03 04 05 06 07 08 09 10 11 12
13 14 15 16 17 18 19 20 21 22 23 24

After first encryption the order becomes:

02 08 14 20 05 11 17 23 01 07 13 19
06 12 18 24 03 09 15 21 04 10 16 22

After the second encryption the order comes:

08 23 12 21 05 13 03 16 02 17 06 15
11 19 09 20 14 01 18 04 20 07 24 10

Are there any obviously obversvable patterns?

Summary of Transposition and Substitution Ciphers

- Transposition ciphers on their own offer no practical security
- But combining transposition ciphers with substitution ciphers, and repeated applications, practical security can be achieved
- Modern symmetric ciphers use multiple applications (rounds) of substitition and transposition (permutation) operations