

Deciphering of historic letters



Deciphering a directory-based encryption

Maxence MONCEL & Loïc ROUQUETTE
EPITA - Laboratoire de Recherche de l'EPITA (LRE)
16 juillet 2026

01

Context

02

Perplexity

03

Hill-Climbing

04

Human-In-The-Loop

05

Conclusion

06

Opening

A series of thin, light blue lines forming a complex geometric pattern of triangles and polygons in the bottom-left corner of the slide.

01

Context

- Ciphred letters from the *XVIII* – *XIXth* century
- Correspondence letters with French colonies in America
- Letters are part of a bigger collection
- The collection also contains some of the keys
- Some letters are already deciphered[Pierrot et al., 2023]

134. 127. 339. 39. les plus complètes 193. 338. 300. 14.
 424. 385. 268. 339. 39. 497. 39. 414. 580. 508. 108.
 139. 301. 841. 149. 631. je m'empresse de mettre sous vos
 yeux 500. 179. 385. 194. 351. 600. 668. 385. 754. 805.
 809. 685. 189. 416. 473. 202. 65. 721. 86. 365. 830. 385.
 540. 804. 449. 561. votre attention.
 685. 847. 523. 39. 429. Je n'ai pas avantageusement
 servi Monsieur, 265. 841. 708. 754. 239. Pour les
 procurer des avis de ce qui se passe 846. 142. 258. 500.
 462. 500. 555. 593. 225. 677. 429. 8. 420. 194. 118. infini-
 ment 68. 754. 563. 420. 80. 39. 685. 628. 672. 385. 194.
 351. clare, qu'il a 25. 14. 590. 193. 685. 600. 222. 190. il a
 dans une plume mienne 457. 500. 62. 638. 385. 418. 476.
 225. 453. 638. 27. hier, et m'en apporte des nouvelles -
 donc je m'empresse de vous rendre compte. ARCHIVES
NATIONALES
 841. 258. 568. 211. 417. 426. J'y enlève le Premier
 385. 194. 810. 125. 301. 437. 131. 704. 508. 452. 344. 408.
 301. 755. 351. 846. 754. 301. 29. 239. 45. 844. 258. 27.

Figure 1: A section of the letters we are trying to decipher. Please notice that plaintext is inserted between two cyphertext sections

- Exclusively substitutions ciphers[Lucks, 1988]
- A mix between nomenclator and homophonic[Dhavare et al., 2013] substitution ciphers cipher
- Also called *directory-based encryption*



- Exclusively substitutions ciphers[Lucks, 1988]
- A mix between nomenclator and homophonic[Dhavare et al., 2013] substitution ciphers cipher
- Also called *directory-based encryption*
- Nomenclator -> Simple substitution but plaintext symbols can be syllables and words

- Exclusively substitutions ciphers[Lucks, 1988]
- A mix between nomenclator and homophonic[Dhavare et al., 2013] substitution ciphers cipher
- Also called *directory-based encryption*
- Nomenclator -> Simple substitution but plaintext symbols can be syllables and words
- Homophonic substitution -> Simple substitution but multiple cypher symbols for a single plaintext symbol

| | | | |
|---------------------------------|---------------------------|-----------------------------|--|
| 1. présent 590 | 1. toujours 380 | N O M S
de
L I E U X. | |
| 2. présent 584 | 2. tout 371 | | |
| 3. prétexte 786 | 3. tout 342 | | |
| 4. principal, aux 451 | 4. trahi 676 | | |
| 5. prochain 921 | 5. traite 565 | Amérique 810 | |
| 6. progrès 822 | 6. transp 537 | | |
| 7. projet 194 | 7. trans 544 | | |
| 8. prompt 633 | 8. trop 661 | | |
| 9. propos 911 | 9. trou 67 | Angleterre 709 | |
| 10. protes 644 | 10. troupe 774 | Anglois 105 | |
| 11. provision, s 295 | 11. troupe 805 | Boston 294 | |
| 12. publi 721 | | Bayonne 495 | |
| 13. puis, que 734 | | Bordeaux 714 | |
| | | Brux 595 | |
| | | Canada 804 | |
| Qua 505 | Va 521 | Cap 5 | |
| que 114 | ve, s 76 | Cap d'Antoine 115 | |
| qui 42 | vi 552 | Cayenne 104 | |
| quo 824 | vo 577 | Charleston 205 | |
| qu 765 | va 564 | Chazao 2 | |
| qualité 574 | van, t 488 | Danmark 504 | |
| quantité 773 | vais 77 | Danica 408 | |
| quand, t 264 | vaissau, x 564 | Espagne 505 | |
| quartier 675 | venir 496 | Espagnole 724 | |
| quel, le 522 | vent 552 | Europe 841 | |
| quodque, s 654 | ver 282 | France 94 | |
| quer 164 | véri 651 | Francisc 185 | |
| question 622 | vérifi 451 | Gorce 204 | |
| qu'il, s 282 | vou 282 | hollande 595 | |
| quintaux 821 | vient 425 | hollandaise 414 | |
| quod 791 | vile 687 | Ile de Cuba 795 | |
| qu'on 812 | vivres 655 | Ile de Re 8 | |
| qu'en 722 | un, e 495 | Ile Royale 200 | |
| | ui 686 | Ile Turque 98 | |
| | somme 82 | la Dominique 500 | |
| | son, t 665 | | |
| | stat 566 | | |

Figure 2 : A section of the key used to cypher the text. Please note the section reserved for the name of places and for null cypher symbols

- Collection of old ciphered texts and (sometimes) the associated keys
- Collection of tools to help deciphering (Cryptool2)
- Helping historians decipher historic texts



Figure 3 : The Cryptool2 logo that was taken from the official Cryptool2 website

A series of thin, light blue lines forming a complex geometric pattern of triangles and polygons, located in the bottom-left corner of the slide.

02

Perplexity

"Does this text make sense?"



"Does this text make sense?"

- A scoring function evaluating a key by its ability to decipher french text that makes sense
- An optimization function will optimize this scoring function



"Does this text make sense?"

- A scoring function evaluating a key by its ability to decipher french text that makes sense
- An optimization function will optimize this scoring function
- Based on one of Markov chains applications (n-grams [[Megyesi et al., 2023](#)])
- First a training phase, then we can compute perplexity

"Does this text make sense?"

- A scoring function evaluating a key by its ability to decipher french text that makes sense
- An optimization function will optimize this scoring function
- Based on one of Markov chains applications (n-grams [[Megyesi et al., 2023](#)])
- First a training phase, then we can compute perplexity
- Show how *"perplex"* is the program towards the given text

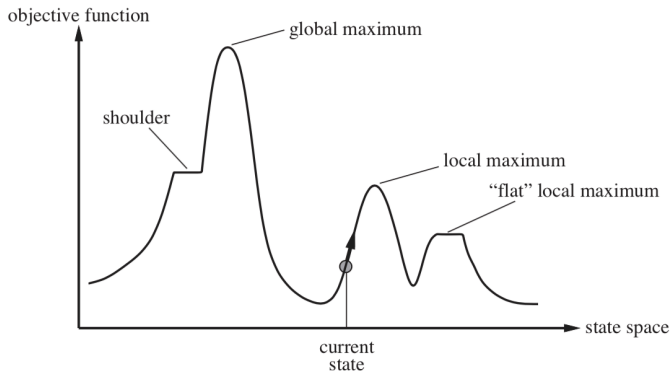


Figure 4 : An example of a scoring function that we are trying to minimize/maximize the value of. There are a lot of pitfalls that hinder our ability to find the global minimum/maximum. Source : <https://www.cs.cmu.edu/15281-s23/coursenotes/localsearch/images/utilitygraph.png>

How to abstract from the plaintext symbols?



How to abstract from the plaintext symbols?
Answer : Encapsulate them in a class and give them a name!



How to abstract from the plaintext symbols?
Answer : Encapsulate them in a class and give them a name!

- Differentiate them from plain strings
- Ensure structure and coherence within the deciphering process (1 token $\leftrightarrow$ 1 cypher symbol)



How to abstract from the plaintext symbols?
Answer : Encapsulate them in a class and give them a name!

- Differentiate them from plain strings
- Ensure structure and coherence within the deciphering process (1 token $\leftrightarrow$ 1 cypher symbol)

Hypothesis

Ciphering by hand is a time-consuming process, thus we assumed that the writers always used the longest plaintext symbol available.

"What happens next depends only on the state of affairs now"



"What happens next depends only on the state of affairs now"

$\forall n \geq 0, (i_0, \dots, i_{n-1}, i, j) \in E^{n+2}$ such as $\mathbb{P}(X_0 = i_0, X_1 = i_1, \dots, X_{n-1} = i_{n-1}, X_n = i) > 0$,

$\mathbb{P}(X_{n+1} = j | X_0 = i_0, X_1 = i_1, \dots, X_{n-1} = i_{n-1}, X_n = i) = \mathbb{P}(X_{n+1} = j | X_n = i).$

With E , the set of all possible states



"What happens next depends only on the state of affairs now"

$\forall n \geq 0, (i_0, \dots, i_{n-1}, i, j) \in E^{n+2}$ such as $\mathbb{P}(X_0 = i_0, X_1 = i_1, \dots, X_{n-1} = i_{n-1}, X_n = i) > 0,$

$\mathbb{P}(X_{n+1} = j | X_0 = i_0, X_1 = i_1, \dots, X_{n-1} = i_{n-1}, X_n = i) = \mathbb{P}(X_{n+1} = j | X_n = i).$

With E , the set of all possible states

How about n-grams?

A set of n consecutive tokens in the text.

A Markov chain of order $n - 1$ as we want to find the probability of the last token in the n -gram considering all the others.

Ex : Probability of 4-gram *["bon", "jour", "a", "tous"]* is the probability of the tokens *["bon", "jour", "a"]* to be followed by the token *"tous"*.

- Training on text dealing with the same subjects, same time period and in the same language as the cyphertext [Megyesi et al., 2023]
- Simply computing probabilities for all possible n-grams in the training texts and saving them



- Training on text dealing with the same subjects, same time period and in the same language as the cyphertext [Megyesi et al., 2023]
- Simply computing probabilities for all possible n-grams in the training texts and saving them

$$\log\left(\frac{\text{times ngram appeared}}{\text{amount of ngrams}}\right)$$



- Training on text dealing with the same subjects, same time period and in the same language as the cyphertext [Megyesi et al., 2023]
- Simply computing probabilities for all possible n-grams in the training texts and saving them

$$\log\left(\frac{\text{times ngram appeared}}{\text{amount of ngrams}}\right)$$

- Computing the mean average probability of all n-grams in the deciphered text (naive approach)
- Penalty if we haven't learned a specific n-gram

Perplexity - Naive approach results

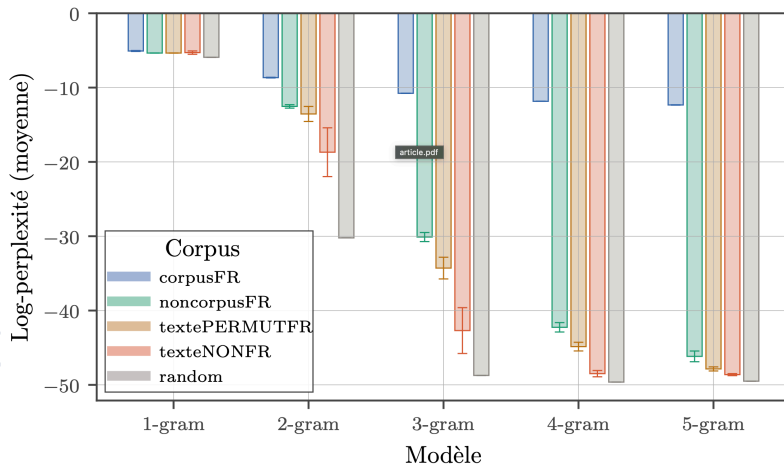


Figure 5 : A table showing the results of our naive perplexity approach on test text of roughly a paragraph of length (about 60 words each)

03

Hill-Climbing



We now can detect French text on the topic of the letter we are trying to decipher
Now we *just* have to find the key



We now can detect French text on the topic of the letter we are trying to decipher
Now we *just* have to find the key

- Multiple approaches
- Initialize the key and get searching
- Need a big enough text sample to ensure correct deciphering



We now can detect French text on the topic of the letter we are trying to decipher
Now we *just* have to find the key

- Multiple approaches
- Initialize the key and get searching
- Need a big enough text sample to ensure correct deciphering

Be careful!

We are going to manipulate a key with 1000 elements.
It's like trying to optimize a 1000 parameters function...

- We know most of the plaintext symbols
- There are plaintext symbols for places and persons names



- We know most of the plaintext symbols
- There are plaintext symbols for places and persons names
- For now we will ignore them and consider them as null symbols
- We just shuffle all the plaintext symbols and add enough null symbols to have 1000 elements in the key

Hill-Climbing - Naive hill climbing[Russell et Norvig, 2020]

- Most basic and naive hill climbing
- Swap 2 elements of the key and keep it if the perplexity score improved
- Used as a baseline to compare to, not as an actual solution



- Most basic and naive hill climbing
- Swap 2 elements of the key and keep it if the perplexity score improved
- Used as a baseline to compare to, not as an actual solution

Swap intelligently

We still make sure to avoid objectively useless swaps such as swapping the same elements or two elements not used in the cyphertext.

Hill-Climbing - Taboo hill climbing

- Naive hill-climbing but with a *taboo list*
- If a swap is unsuccessful, add it to the *taboo list* so we may avoid it



Hill-Climbing - Taboo hill climbing

- Naive hill-climbing but with a *taboo list*
- If a swap is unsuccessful, add it to the *taboo list* so we may avoid it
- Helps converging faster to local maximum than the naive approach
- Hyperparameters that can influence the outcome of the algorithm (number of iterations, max size of taboo list)



Hill-Climbing - Taboo hill climbing

- Naive hill-climbing but with a *taboo list*
- If a swap is unsuccessful, add it to the *taboo list* so we may avoid it
- Helps converging faster to local maximum than the naive approach
- Hyperparameters that can influence the outcome of the algorithm (number of iterations, max size of taboo list)

Taboo "list"?

The taboo list works more like a FIFO structure rather than an actual list because it makes no sense to forbid a swap that used to be bad several iterations ago. If it is full, the oldest taboo is removed.

| gram size | score |
|-----------|--------|
| 3 | -42.4 |
| 4 | -49.1 |
| 5 | -49.52 |

Figure 6 : The results of the naive hill-climbing depending on the size of the n-grams



| gram size | score |
|-----------|--------|
| 3 | -42.4 |
| 4 | -49.1 |
| 5 | -49.52 |

Figure 6 : The results of the naive hill-climbing depending on the size of the n-grams

| gram size | score |
|-----------|--------|
| 3 | -41.21 |
| 4 | -49.15 |
| 5 | -49.48 |

Figure 7 : The results of the naive hill-climbing with taboo list depending on the size of the n-grams

04

Human-In-The-Loop



This is the part where we take care of the plaintext symbols referring to the names of places.

- Not yet tested! Only implemented
- The algorithm is asking for help
- Allows us to escape local maximums
- Allows us to manage the case of plaintext symbols referring to place names

This is the part where we take care of the plaintext symbols referring to the names of places.

- Not yet tested! Only implemented
- The algorithm is asking for help
- Allows us to escape local maximums
- Allows us to manage the case of plaintext symbols referring to place names

Cross disciplinary research

We will need the help of the LRE's human and social researchers in order to find the places these letters are referring to.

- Adding new plaintext tokens



- Adding new plaintext tokens
- Checking the current key



- Adding new plaintext tokens
- Checking the current key
- Checking the deciphered text with the current key



- Adding new plaintext tokens
- Checking the current key
- Checking the deciphered text with the current key
- Making manual swaps

| | | | | |
|-----------------|-------------------|-----------------|-----------------|-----------------|
| 185: gu | 186: marchandises | 187: cer | 188: informer | 189: gri |
| 190: espagnols | 191: ai | 192: cote | 193: gle | 194: sur |
| 195: bo | 196: guerre | 197: fro | 198: livre | 199: sti |
| 200: rai | 201: subsistances | 202: ??? | 203: envoi | 204: sieur |
| 205: est | 206: publi | 207: er | 208: celles | 209: pru |
| 210: curacao | 211: de | 212: tres | 213: mi | 214: ??? |
| 215: ils | 216: sans | 217: negociants | 218: provisions | 219: cli |
| 220: ??? | 221: se | 222: cent | 223: ki | 224: etre |
| 225: tion | 226: vos | 227: soin | 228: eaux | 229: sion |
| 230: sucre | 231: actions | 232: charleston | 233: utilite | 234: ob |
| 235: t | 236: ??? | 237: ??? | 238: ??? | 239: expedi |
| 240: ga | 241: mission | 242: ??? | 243: ent | 244: e |
| 245: xe | 246: xi | 247: ses | 248: elle | 249: voyage |
| 250: iers | 251: ran | 252: ier | 253: depeche | 254: ??? |
| 255: vaiseau | 256: sont | 257: trant | 258: sauvages | 259: ministere |
| 260: chi | 261: ??? | 262: generaux | 263: ??? | 264: sept |
| 265: ien | 266: combat | 267: ance | 268: somme | 269: oient |
| 270: hostilites | 271: la vera eux | 272: terre | 273: que | 274: angleterre |

Figure 8 : A first look at the Human-In-The-Loop feature displaying the key

```

468|823| 179 | 83 |779| 84| 46 |399| 501 | 49 | 399|716|571|747| 411|200|275|492|614|460|495|423| 566 | 827| 12 |254|842| 181 |460
ro co mauvais temps et w tour it bruit st vincent it fo no qu voir rai car i il a ri m forbans cafe offr ??? mes question a

55 |273|496|113|480|527|221|460|495|406|657|273| 16 | 825| 85|527|273|716| 487 |551| 437 |444|813| 394 |194| 725 | 435| 434|195|147
present que s ou i gne se a ri e po que partage leur di gne que fo impossible or sentir ??? u la georgie sur grand reso deux bo p

48| 97|
l y Arrivé depuis quatre jours dans la colonie je ne puis Monseigneur vous rendre qu'un compte très sommaire de l'état où je la trouve. Il, paraî

-1
t qu'il y regne de l'ordre et de la tranquillité a ri e co ba me ne ri bo voyage si pi dans ??? gu co leurs na quo lors gna ja qu ab

282|495|147| 82|749|594|113|215|460|495|310|837|453|594|495| 839| 629 | 661 | 186 |352|779| 14 | 153| 489| 550 | 225|779| 843 | 540
a ri p ti en ne ou ils a ri o ba me ne ri vant ordinaire vouloir marchandises gna et utile fort nous danoix tion et encore toulon

```

Figure 9 : A first look at the Human-In-The-Loop feature displaying the current text being deciphered

A series of thin, light blue lines forming a complex geometric pattern of triangles and polygons in the bottom-left corner of the slide.

05

Conclusion

- Perplexity method to judge quality of deciphered text



- Perplexity method to judge quality of deciphered text
- First simple hill-climbing algorithms (will have to do better than that)



- Perplexity method to judge quality of deciphered text
- First simple hill-climbing algorithms (will have to do better than that)
- First steps for Human-In-The-Loop features



- Perplexity method to judge quality of deciphered text
- First simple hill-climbing algorithms (will have to do better than that)
- First steps for Human-In-The-Loop features

And also...

My predecessor on this subject (baptiste.cormoran) had already done some work that I documented.

Abstract geometric lines in the bottom left corner, forming a series of connected triangles and polygons.

06

Opening

- Improve hill climbing (Simulated annealing)
- Play around with Hyperparameters



- Improve hill climbing (Simulated annealing)
- Play around with Hyperparameters
- How "bad" of a hill-climbing algorithm can we get away with Human-In-The-Loop?
- Locking parts of the key during Human-In-The-Loop to ease deciphering



- Improve hill climbing (Simulated annealing)
- Play around with Hyperparameters
- How "bad" of a hill-climbing algorithm can we get away with Human-In-The-Loop?
- Locking parts of the key during Human-In-The-Loop to ease deciphering
- Improving perplexity computation ("smoothing" and using various n-grams sizes)

- Improve hill climbing (Simulated annealing)
- Play around with Hyperparameters
- How "bad" of a hill-climbing algorithm can we get away with Human-In-The-Loop?
- Locking parts of the key during Human-In-The-Loop to ease deciphering
- Improving perplexity computation ("smoothing" and using various n-grams sizes)
- OCR for full deciphering pipeline[Yin et al., 2019]?

Thank you for your attention



- Dhavare, A., Low, R. M., & Stamp, M. (2013)
Efficient cryptanalysis of homophonic substitution ciphers.
Cryptologia.
- Lucks, M. (1988)
A constraint satisfaction algorithm for the automated decryption of simple substitution ciphers.
Conference on the Theory and Application of Cryptography.
- Megyesi, B., Esslinger, B., Fornés, A., Kopal, N., Láng, B., Lasry, G., de Leeuw, K., Pettersson, E., Wacker, A., & Waldispühl, M. (2020)
Decryption of historical manuscripts : the DECRYPT project.
Cryptologia.
- Megyesi, B., Sikora, J., Fornmark, F., Waldispühl, M., Kopal, N., & Mikhalev, V. (2023)
Historical Language Models in Cryptanalysis : Case Studies on English and German.
International Conference on Historical Cryptology.
- Pierrot, C., Desenclos, C., Gaudry, P., & Zimmermann, P. (2023)
Deciphering Charles Quint (A diplomatic letter from 1547). In C. Dahlke & M. Göggerle (Éd.), *Linköping Electronic Conference Proceedings*.

- Russell, S., & Norvig, P. (2020)
Artificial Intelligence : A Modern Approach (4th Edition)
.Pearson.
- Yin, X., Aldarrab, N., Megyesi, B., & Knight, K. (2019)
Decipherment of historical manuscript images.
2019 International Conference on Document Analysis and Recognition (ICDAR).