Réseaux et sécurité
M1 Informatique

Alexis
PLESSIER

TP sur l'implémentation et la cryptanalyse
Du chiffrement de Vigenère

I. Partie I – Implémentation du chiffrement de Vigenère

J'ai décidé d'implémenter le chiffrement de Vigenère avec le langage **Python** car c'est le langage que je maitrise le plus, avec lequel je me sens le plus à l'aise et il ne présente aucune difficulté à s'accommoder au problème présenté.

Pour le chiffrement rien de plus simple : je converti les caractères de ma chaîne de caractère en nombre décimal selon la table ASCII et je renvoie le résultat de la somme du caractère du message à chiffrer et du caractère de la clé, le tout modulo 26. Tout en sachant qu'avant cette étape, le texte à été converti en minuscule et les caractères différents de l'alphabet ont été supprimé, je peux renvoyer le message chiffrer sans crainte.

II. Partie II – Cryptanalyse par estimation de la longueur de la clé et analyse fréquentielle

a. Méthode de Babbage et Kasiki

Cette partie m'a pris plus de temps que prévu, notamment pour chercher toutes les occurrences de séquences de 3 lettres ou plus qui se répètent. J'ai décidé de mettre toutes les séquences possibles dans un tableau (si la clé a une longueur supérieure à 3) et ensuite de les compter en les implémentent dans un dictionnaire qui est une structure adaptée à la situation. Une fois toutes mes séquences et leurs apparitions, je ne garde que celles apparaissant deux fois ou plus.

Avec les séquences les plus redondantes, je vais pouvoir rechercher dans le texte chiffré leurs indices d'apparitions et, ainsi, calculer leurs distances d'apparitions quelques soit leur nombre d'occurrences. Pour améliorer ce procédé, je peux supprimer 10% des occurrences qui apparaissent le moins pour me rapprocher d'un résultat plus sûr. Je sais que la longueur de la clé est le plus grand dénominateur commun à toutes ces distances, je cherche donc le PGCD de toutes ces distances et je renvoie le résultat qui sera la longueur de la clé utilisée pour le chiffrement du message.

b. Test de Friedman

La probabilité Kr que deux lettres minuscules prisent aléatoirement dans un texte soient similaires est défini par le nombre de lettre dans l'alphabet et est égale à $\frac{1}{26}$ = 0.038.

L'indice Kr d'un texte français est environ égal à **0.0778**.
On sait que Ke, l'indice de coïncidence d'un texte en Anglais, est environ égal à **0.067** et c'est normal qu'il soit diffèrent de celui de la langue française car les mots des deux langues ne sont pas les mêmes et donc les lettres n'apparaissent pas à la même fréquence. Et encore, les deux langues ont le même alphabet (latin), mais le russe qui utilise l'alphabet cyrillique possède 33 lettres et donc plus de combinaisons possède et moins de possibilités d'avoir deux lettres aléatoires similaire (indice de coïncidence les plus bas au monde avec **0.0529**).

Pour implémenter cet indice, je vais simplement chercher le nombre d'occurrence de chaque caractère ainsi que la longueur du texte dont ils sont issus pour leur appliquer la formule ci-dessous et les additionner afin d'obtenir le résultat propre à mon texte. **Attention**, comme dis ci-dessus, ce programme donne l'indice de coïncidence pour les textes utilisant l'alphabet latin seulement.

Formule de l'indice de coïncidence :

$$\sum_{i=a,\dots,z} \frac{n_i(n_i-1)}{n(n-1)}$$

Avec n_i = occurrence du caractère i

Avec n = la longueur du texte Tr

La longueur de la clé est donc implémentée par la formule suivante :

$$L = \frac{(Ke-Kr)}{(K-Kr)}$$

Avec Ke = 0.067
Avec Kr = 1/26
Avec K = Indice incidence du texte

On peut remarquer que La longueur de la clé dépend surtout de la l'indice de coïncidence du texte entré en paramètre. Ce dernier est très sensible à la longueur du texte, ce qui est une variable sensible au résultat final, il faut donc faire attention que la longueur est suffisante pour que le résultat final ne soit pas trop impacté.

c. Analyse fréquentielle

Pour l'analyse fréquentielle, j'ai initialisé un tableau de 26 cases à 0 et ensuite compté les occurrences de chaque lettre pour détecter quelles lettres apparaissaient le plus. J'ai effectué ce procédé sur n sous-textes où n est la longueur de la clé. On considérant que E est la lettre la plus fréquente, on effectue le décalage entre la lettre qui apparait le plus et E et on l'ajoute aux autre lettres pour obtenir la clé probable.

III. Partie III – Cryptanalyse par méthode du mot probable

Cette dernière partie repose sur l'analyse d'un texte chiffré par un mot probable. Il faut donc dans un premier temps soustraire ce mot probable au texte chiffré à chaque position possible. Une fois notre liste de mots de même longueur, notre clé probable se trouve à travers toutes ces combinaisons, il faut donc chercher une répétition à travers toutes ces possibilités. Les mots se répétant sont fortement probable d'être clé du chiffrement.

IV. Conclusion

Le chiffrement de Vigenère est un chiffrement polyalphabétique qui fût cassé en 1863 après plus de 300 ans d'invulnérabilité. Il a fallu attendre Kasiski et sa méthode d'estimation de la longueur de la clé pour commencer à venir à bout de ce chiffrement. On se rend également compte qu'il est aujourd'hui, et grâce aux avancées technologiques, très facile de déchiffrer un texte quand il aura fallu 300 ans il n'y a pas si longtemps que ça. Il a également était rendu possible avec l'avancée mathématiques de ces dernières années où l'on peut évidemment émettre un lien indéniable entre les chiffrements et les mathématiques.