Quebra da cifra de Vigènere utilizando o método de Friedman

Marcelo B. de Azevedo

[1]Engenharia de Software – Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
Av. Ipiranga, 6681 – 90.619-900 – Porto Alegre – RS – Brasil

marcelo.bernardy@acad.pucrs.br

1. Introdução

Dentro da disciplina de Segurança de Sistemas foi proposto, como primeiro trabalho, o desenvolvimento de um algoritmo visando a quebra da cifra de Vigènere.

A cifra pode ser definida como um método de criptografia que usa uma série de diferentes cifras de César baseadas em letras de uma senha. Na cifra de César cada letra do alfabeto é deslocada em uma mesma distância de posições. Já na cifra de Vigènere o deslocamento tende a ser diferente para a mesma letra, tendo como base a chave da mensagem definida.

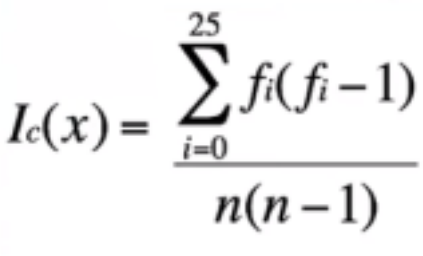
O desenvolvimento do trabalho foi realizado dentro da linguagem *open source* e criada pela Google "Go". Podemos aferir de forma resumida que o algoritmo se divide em: recebimento do arquivo com a mensagem cifrada; verificar o tamanho da chave com base no cálculo do Índice de Coincidência da língua portuguesa; exibir as opções disponíveis de chave para que se possa escolher a chave mais condizente; decodificar com a chave proposta e armazenar a saída em um novo arquivo de texto com o conteúdo em texto claro.

2. Cálculo do tamanho da chave

Para realizar o cálculo do tamanho da chave foi considerado que o texto claro esteja escrito em Português e com o Índice de Coincidência da língua igual a 0.072723.

O passo a passo do algoritmo consiste em primeiramente contabilizar a frequência de cada letra sobre o conteúdo do texto completo e após aplicar sobre a função descrita na imagem no final desta seção. Caso o primeiro conjunto com o texto completo não fique próximo ao valor do Índice de Coincidência é realizado uma divisão do texto em duas partes. A primeira parte com os carácteres seguindo uma ordem da posição sobre o texto original cifrado [0,2,4,6,...] e a segunda parte seguindo uma ordem [1,3,5,7,...] até o fim da mensagem. Após essa mensagem dividida é realizada outra vez o cálculo para cada mensagem, verificado se elas estão próximas ao Índice, caso não, o processo de divisão é realizado mais uma vez agora dividindo o texto em 3 partes e assim sucessivamente até encontrar um Índice de Coincidência próximo.

Caso o cálculo esteja em um valor aproximado ao da língua, com uma margem de 0.3, o tamanho da chave é o respectivo número no qual a mensagem foi dividida. Se o número de divisões do texto cifrado chegar a dez tentativas o algoritmo é encerrado sem calcular o tamanho da chave.



$$I_c(x) = \frac{\sum_{i=0}^{25} f_i(f_i - 1)}{n(n-1)}$$

Figura 1. Fórmula do cálculo do Índice de Coincidência.

A figura acima *fi* representa a frequência de cada letra, sendo *i* de 0 a 25, e *n* o número total de carácteres do texto cifrado.

3. Decifrar a chave

Após encontrado o tamanho da chave, a mensagem cifrada é dividida em pequenas partes de tamanho igual ao da chave. Depois agrupamos novamente o texto pelo respectivo índice de cada parte da mensagem, ou seja, pensando em um exemplo de chave sete, dividimos o texto cifrado em sete partes e após agrupamos a primeira posição de cada parte da mensagem em uma nova *substring*, após repetimos o processo para a segunda posição e assim sucessivamente até o tamanho da chave. Cabe ressaltar que assim organizamos o texto de maneira na qual cada *substring* foi cifrado com o mesmo caractere da chave.

Com a divisão realizada é feito um cálculo da frequência para cada *substring* e após é realizado o cálculo da diferença em módulo entre o caractere mais frequente da língua, no português ”A”, e o caractere mais frequente desta parte da mensagem. Desta maneira a diferença representa a posição do caractere em que foi cifrado a letra ”A”, sendo que para as posições podemos considerar [A = 0, B = 1, C = 2, ...]. Como a língua portuguesa possui 2 caracteres com o nível de Índice de Coincidência muito próximos, o mesmo processo se repete para a letra ”E”, e após é exibido para usuário uma opção de palpite para a chave condizente, como mostrado na figura abaixo:

```
Digite a possível chave escolhendo uma das duas letras apresentadas em cada palavra: ex: avelino

-> azipmro
-> wvelink
```

Figura 2. Algoritmo listando as prováveis chaves

4. Decifrar o texto

Com a chave em texto claro encontrada, o processo de decifrar o texto é simples. Basta aplicar a fórmula *Pi = (26 + Li - Ki) mod 26* para cada caractere no qual a *substring* foi cifrada no mesmo caractere, processo explicado na sessão anterior.

Na fórmula *Li* representa o índice da letra que está sendo decifrada; *Ki* representa o índice da letra na chave; 26 o número total de caracteres do Alfabeto.

Os trechos abaixo representam respectivamente uma parte do texto cifrado e decifrado do livro ”A Democracia”, escrito por Jaime de Magalhães de Lima:

Texto Cifrado: *qpixpnqiigzmahavrywfhiqidarocjvloradztfjywcvvfuywvmsnwzcoyidczbemqlqasszvtiwilbeowiwsdsyiewojylxnwxjrll*

Texto Decifrado (separando as palavras com espaços): *quem ha cincoenta annos tivesse a coragem de publicar um livro como o de sumner maine seria julgado visionario ou apaixonado*

5. Conclusão

Com o algoritmo descrito ao decorrer deste trabalho quebramos a cifra de Vigènere, uma cifra que demorou séculos a ser quebrada.

Podemos ressaltar que foi utilizado método de Friedman com o uso do Índice de Coincidência. Mesmo gerando uma distribuição uniforme em análises de frequência, essa cifra tem uma vulnerabilidade: a palavra-chave é usada várias vezes em um texto grande. Podemos ter então como lição que que um texto cifrado não deve possuir qualquer padrão de repetição identificável, para que assim, não haja distinção de um texto completamente aleatório.

6. Referências

Cifra de Vigenère, 2021. Disponível em:

https://wiki.imesec.ime.usp.br/books/criptografia/page/cifra-de-vigenere

Acesso em: 22 de abril de 2021.

Letter frequency. WIKIPEDIA, 2021. Disponível em:

https://en.wikipedia.org/wiki/Letter_frequency.

Acesso em: 23 de abril de 2021.