860 Analisador de entropia textual

Claude Shannon foi um matemático e cientista da computação que nasceu em 30 de abril de 1916 e morreu em 24 de fevereiro de 2001. Foi ele quem criou as fundações matemáticas que definiram as regras gerais da teoria da informação moderna. Em seu paper de 1948, *Uma Teoria Matemática da Comunicação,* uma medida chamada entropia foi proposta: um grau de indeterminação associado com uma fonte aleatória sem memória. Nós estamos interessados aqui no uso do conceito de entropia para analisar textos no nível de sua variedade de palavras.

Definimos a entropia de um texto T, com λ palavras e n palavras distintas pela fórmula

$$E_T(p_1,\ldots,p_n)=\frac{1}{\lambda}\sum_{i=1}^{n} p_i[log_{10}(\lambda)-log_{10}(p_i)]$$

Onde $p_i, i=1,\ldots,n$, é a frequência de cada i -palavra no texto T, ou seja, p_i é o número de vezes que a i -palavra ocorre em dado texto. Se considerarmos que um texto de cumprimento λ (um texto com λ palavras) é tão rico quanto mais n diferentes palavras e, entre os textos com a mesma quantidade λ de palavras e a mesma quantidade n de palavras distintas, é mais rico que os textos em que as palavras variam menos em frequência, podemos concluir que a entropia é com certeza uma medida muito útil para comparar a riqueza de dois ou mais textos. Para comparar textos com número diferente de palavras λ , temos o que podemos considerar uma "entropia relativa" E_{rel} , definida como o consciente entre a entropia E_T do texto e a entropia máxima E_{max} , e multiplicando por 100 se desejamos uma porcentagem:

$$E_{rel}=\frac{E_t}{E_{max}}\times 100$$

A máxima entropia E_{max} é apenas a entropia de um texto com o mesmo λ de palavras e no qual cada palavra ocorra exatamente uma vez $(i.e., n:=\lambda, p_i:=1)$:

$$E_{max}=\frac{1}{\lambda}\sum_{i=1}^{\lambda} 1\cdot[log_{10}(\lambda)-log_{10}(1)]=log_{10}(\lambda)$$

Dado um texto T, escreva um programa que compute o total número λ de palavras em T, a entropia E_T do texto, e sua entropia relativa E_{rel} . Para determinar os resultados, seu programa deve considerar letras maiúsculas e minúsculas da mesma forma (por exemplo, as palavras “Casa”, “casa” e “CASA” devem ser consideradas iguais). Além disso, no contexto desse programa, uma palavra é uma sequência consecutiva de caracteres diferentes das sinais

de pontuação , . : ; ! ? “ ( ) assim como espaços, tabs e caracteres para pular linha (‘\n’). Palavras com somente uma letra devem ser consideradas.

Entrada

A entrada contém diversos textos T, cada um necessariamente com mais de uma palavra $(\lambda>1)$. Você pode assumir que o tamanho máximo das palavras é de 20 caracteres e que um único texto não possui mais de 100.000 palavras.

Uma linha contendo apenas “******END_OF_TEXT******” representa o final de cada texto, e uma linha contendo “******END_OF_INPUT******” representa o final da entrada. Você pode ter certeza que essas palavras reservadas não aparecerão dentro de um texto. Além dessas palavras, qualquer elemento pode aparecer em um texto, incluindo linhas em branco.

Saída

Na saída, escreva uma linha para cada teste, cada uma contendo três números: o primeiro com o total número λ das palavras em T; o segundo com a entropia E_T do texto arredondada para um dígito decimal; e o último com a entropia relativa E_{rel} , em porcentagem, e arredondada para ser um inteiro.

Nota: O trecho abaixo é de “*Memória*”, de *Barbra Streisand* (dois primeiros versos).

Exemplo de Entrada

Midnight, not a sound from the pavement
Has the moon lost her memory?
She is smiling alone
In the lamplight, the withered leaves collect at my feet
And the wind begins to moan
******END_OF_TEXT******
Memory, all alone in the moonlight
I can dream of the old days
Life was beautiful then
I remember the time I knew what happiness was
Let the memory live again
******END_OF_TEXT******
******END_OF_INPUT******

Exemplo de Saída

33 1.4 93

31 1.3 89