

1 Entropy of the book in python

I applied entropy to the first chapter of Dross's book.

```
import math
from collections import Counter
import pandas as pd

def shannon_entropy(p):
    if p == 0:
        return 0
    return -p * math.log2(p)

def calcular_entropia(texto_cifrado):
    count_words = pd.Series(Counter(texto_cifrado.split()))
    total_words = len(texto_cifrado.split())
    entropy = 0

    for word, count in count_words.items():
        p = count / total_words
        entropy += shannon_entropy(p)

    return entropy

def cifradoCesar(tx):
    alfabeto = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
    cifrado = ""
    clave = 3
    for letra in tx:
        if letra in alfabeto:
            indice = alfabeto.find(letra)
            indice += clave
            if indice >= len(alfabeto):
                indice -= len(alfabeto)
            cifrado += alfabeto[indice]
        else:
            cifrado += letra
    return cifrado, calcular_entropia(cifrado)

def cifradoCisar(tx):
    alfabeto = "cisarbdejklnopqrstuvwxyzCISARBDEJKLMNOPQRSTUVWXYZ0123456789"
    cifrado = ""
    palabras = tx.split()
    for palabra in palabras:
        clave = len(palabra)
        for letra in palabra:
            if letra in alfabeto:
                indice = alfabeto.find(letra)
                indice -= clave
                if indice < 0:
                    indice += len(alfabeto)
                cifrado += alfabeto[indice]
            else:
                cifrado += letra
        cifrado += " -"
```

```

return cifrado.strip(), calcular_entropia(cifrado.strip())

def cifradoj_FBI(tx):
    alf = {
        "e" : "ak", "a" : "ow", "o" : "xu", "i" : "er",
        "u" : "vi", "s" : "nm", "n" : "pl", "r" : "ea",
        "0" : "o", "1" : "i", "2" : "u", "3" : "y",
        "4" : "t", "5" : "r", "6" : "e", "7" : "w",
        "8" : "q", "9" : "p", "." : "d", "," : "s"
    }
    cifrado = ""
    tx = tx.lower()
    for letra in tx:
        if letra in alf:
            cifrado += alf[letra]
        else:
            cifrado += letra
    return cifrado, calcular_entropia(cifrado)

filename = '/content/cuento.txt'
with open(filename, 'r') as file:
    content = file.read()

cifrado_cesar, entropia_cesar = cifradoCesar(content)
cifrado_cisar, entropia_cisar = cifradoCisar(content)
cifrado_fbi, entropia_fbi = cifradoj_FBI(content)

print("Entropia-para-Cifrado-Cesar:", entropia_cesar)
print("Entropia-para-Cifrado-Cisar:", entropia_cisar)
print("Entropia-para-Cifrado-FBI:", entropia_fbi)

mayor_entropia = max(entropia_cesar, entropia_cisar, entropia_fbi)

if mayor_entropia == entropia_cesar:
    print("El-cifrado-Cesar-tiene-la-mayor-entropia.")
elif mayor_entropia == entropia_cisar:
    print("El-cifrado-Cisar-tiene-la-mayor-entropia.")
else:
    print("El-cifrado-FBI-tiene-la-mayor-entropia.")

```

1.1 Results

Entropía para Cifrado César: 8.957045486898574

Entropía para Cifrado Cisar: 8.946328001833

Entropía para Cifrado FBI: 8.821032583492665

El cifrado César tiene la mayor entropía.