Performance of mycat

# Luis Ricardo Espinoza Rodríguez A01229141

Para poder calcular el tiempo de ejecución de ambos scripts cree un archivo de Python que ejecutara ambos scripts. Ambos archivos los probé con el archivo test.txt. Este es un archivo txt de la versión de King james de la biblia que descargué del proyecto Gutenberg. <http://www.gutenberg.org/cache/epub/10/pg10.txt>. Este archivo contiene poco menos de 26mil líneas. Y contiene 224mil palabras.

>wc -l test.txt

25974 test.txt

>224512 test.txt

start\_time = time.time()

os.system('cat test.txt > /dev/null')

end\_time = time.time()

print("cat regular se tardo {} milisegundos").format(time\_arr[0])

Lo que hice es calcular la diferencia del tiempo cuando se ejecuto ambos archivos. Aquí estoy ejecutando el comando de *cat* con el archivo test.txt.

start\_time = time.time()

os.system('./cat2 test.txt > /dev/null')

end\_time = time.time()

time\_arr.append((end\_time-start\_time)\*1000)

print("cat2 se tardo {} milisegundos").format(time\_arr[1])

Aquí estoy ejecutando el binario de mi archivo de cat2. Algo extra que hago es mandar el output a null para que no sea tan verbose.

El output en la terminal es el siguiente.

>python cat.py

cat regular se tardo 12.7608776093 milisegundos

cat2 se tardo 11.6050243378 milisegundos

Por esto vemos que cat2 es 9.05% más rápido.

A screenshot of a social media post

Description automatically generated

Por último hacemos plot de ambos tiempos. Esto lo hice mediante matplotlib

plt.plot(time\_arr, '\*', color=[1, 0.5,1])

plt.show()