Programación II

Trabajo práctico final: Texto Predictivo

Agustín López

Facultad de Ciencias Exactas Ingeniería y Agrimensura

1 Programa C

El objetivo principal del programa escrito en C se centró en dos tareas principales:

1. **Leer archivos de la persona proporcionada como argumento:**
   En primer lugar, el programa ejecuta el comando `ls Textos<nombre>.txt`, almacenando su salida en un archivo auxiliar denominado `archivo.txt`. Esta acción tiene como fin conocer los nombres y la cantidad de textos escritos por la persona. Posteriormente, emplea un bucle while para leer línea por línea el archivo, procesando cada texto de manera individual en cada iteración.

2. **Agregar la entrada sanitizada al archivo `Entradas/<nombre>.txt`:**
   En cada iteración del bucle mencionado, se ejecuta la función `agregar_entrada`, la cual incorpora al archivo `Entradas/<nombre>.txt` el contenido sanitizado del texto actual. Esta acción se realiza siguiendo las pautas indicadas en el enunciado del trabajo.

Una vez finalizado el bucle, se invoca al programa escrito en Python, el cual se encuentra listo para operar con el archivo `Entradas/<nombre>.txt`.

En el programa en C, no hubo decisiones particularmente relevantes, ya que como el objetivo era bastante claro y directo, no logré encontrar muchas alternativas para abordarlo.

2 Programa Python

El programa en Python fue diseñado con el propósito principal de trabajar con tres archivos distintos: la entrada, las frases incompletas y la salida.

1. **Carga inicial de datos:**
   La primera acción consistió en cargar las oraciones del archivo de entrada y las frases incompletas en dos listas de strings separadas.

2. **Lógica central en la función `completar_frases`:**
   Esta función posee la lógica central y la decisión más desafiante del programa: el algoritmo para determinar la palabra más probable. Después de investigar, encontré inicialmente el modelo de trigramas, basado en la creación de un diccionario donde las claves eran ternas de strings y los valores indicaban la frecuencia de ocurrencia en el archivo de entrada. Sin embargo, se identificó un problema al tratar la frase “te _ fumabas unos chinos en Madrid”. A pesar de encontrar múltiples instancias donde “te” era el primer elemento de la 3-upla, las palabras “fumabas”, “unos”, “chinos”, “en” y “Madrid” no aparecían en el texto, lo que resultaba en un fallo en el modelo de trigramas.

   Para superar esta dificultad, adapté el modelo y se implementaron 2 duplas, lo que permitió una mejor aproximación al contexto. Al analizar las palabras posteriores a “te” y anteriores a “fumabas”, se creó una estructura donde cada palabra candidata era una clave y el valor representaba la probabilidad de ser la palabra correcta. Finalmente, se seleccionó la palabra con el puntaje más alto para reemplazar el guion bajo en la frase incompleta.