

MODO DE EMPLEO DE LA HERRAMIENTA

La utilización de la herramienta consiste en pasos básicos que incluyen compilar, cargar archivos de texto, ejecutar el programa principal y finalmente graficar las estadísticas obtenidas. A continuación se explica cada uno de los procesos de forma detallada.

A.1. Descargar la carpeta de la herramienta Pattern Recognition.

Con mentalidad de que la persona que quiera acceder a la aplicación tenga derecho a editar y presentar mejorías la herramienta esa licenciada mediante Creative Commons, bajo el tipo de [licencia](#) Attribution-NonCommercial 4.0 International.

El proyecto ha sido colocado en un repositorio digital en GitHub desde donde se lo puede clonar a otra máquina que desee hacer las veces de host para la herramienta.

Para facilidad del usuario, el enlace al proyecto se indica en la siguiente URL:

<https://github.com/robertomaldonado/PatternTool>

Una vez que se ha descargado la carpeta, se debe verificar que dentro de la raíz de la carpeta, existan los archivos *TextParser*, *FileHandler* y *Analyzer* (Con ambas extensiones: *.cpp* y *.h*). De igual forma se debe determinar que existan en el directorio las carpetas llamadas *shells* y *files*. Esto es importante puesto que los resultados de los análisis se almacenarán en *files* y los resultados visuales en *shells*, una vez que todo el proceso se haya cumplido de forma satisfactoria.

A.2. Compilar el código fuente de la herramienta.

Con el propósito de generar la traducción del programa C++ a lenguaje de máquina se utiliza el compilador. (En este caso se emplea el compilador. Al compilar no se debe olvidar de cargar todas las clases vinculadas al programa principal, por lo que la instrucción debe tener algún parecido a la siguiente sentencia:

```
g++ -std=c++11 main.cpp TextParser.cpp Analyzer.cpp FileHandler.cpp -o executable
```

La sentencia es ejecutada por medio de una consola de comandos que debe permanecer abierta dentro del mismo directorio. La aparición de un archivo ejecutable, sucede después de compilar y realizar la acción de enlace con librerías o clases externas que se utilizan dentro del código.

A.3. Cargar archivos de textos para la comparación.

Antes de ejecutar el programa para verificar la similitud entre dos autores distintos, se deben cargar los archivos de forma correcta para que sean accesibles al programa escrito en *C++*. Esto implica colocar el texto obtenido del primer autor (Autor B) en el archivo *default1* con extensión *.txt*. De forma semejante, se debe colocar el contenido de cada uno de los textos del segundo autor (Autor A) en los archivos *default2.txt* hasta *default6.txt*.

A.4. Ejecución del programa principal.

Ahora, desde la consola que fue utilizada para ejecutar la compilación en el paso anterior, se realiza un llamado al archivo ejecutable con el nombre que fue otorgado.

Siguiendo con las instrucciones anteriores, en este caso la sentencia será:

```
./executable
```

Es importante especificar que el archivo es de carácter ejecutable por lo que no se puede omitir la estructura inicial *./* que precede al nombre del mismo.

Cuando el programa entra en ejecución, el usuario deberá seguir las instrucciones del mismo. Los únicos parámetros solicitados son el modo de *debugging* y el número de textos totales que posee el operador de la aplicación. En caso de que el usuario desee realizar pruebas independientes con agrupaciones específicas de parámetros, entonces el programa verificará cuales son los el usuario desea incluir.

En la consola se verán los resultados del análisis, los que son presentados en porcentajes de similitud. Mientras mayor es la similitud, esto quiere decir que la autoría puede atribuirse al mismo autor en un porcentaje de probabilidad. Los resultados quedan abiertos, para que la conclusión sea obtenida por una persona, puesto que el sistema desarrollado no es predictivo.

A.5. Verificar y otorgar los permisos necesarios a !binBash

En el mismo directorio debe habilitarse los permisos de ejecución para lanzar los procesos por medio de instrucciones pre-escritas en un shell script. Para ello se debe hacer la siguiente sentencia:

chmod 777 /shells

sentencia que concede los permisos a todos los archivos que se encuentran en el directorio especificado.

A.6. Ejecución de los comandos para producir componentes visuales

Una vez que los permisos han sido otorgados a los shells y los datos estadísticos han sido escritos se procede al proceso de procesamiento de gráficos. Para ello se debe ingresar en el directorio de *shells* con el comando “*cd shells*”. Como se puede divisar existen varios archivos, los cuales han sido escritos de forma anticipada, puesto que el orden de los parámetros es de extrema importancia. Al ser ejecutable, se debe llamar al archivo *plot.sh* de la siguiente forma:

./plot.sh.

Este comando nos dará el retroalimentación en la consola sobre los procesos que han sido completados. Finalmente estos comandos generan una carpeta denominada *histograms* que contienen los gráficos estadísticos.