UTN.BA

UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

PROCESAMIENTO DEL LENGUAJE NATURAL

TRABAJO PRÁCTICO INDIVIDUAL

CÓDIGO 01: PLAGIO

PROFESOR: Mg. Ing. Hernán Borré

AUTOR: Juan Francisco Paoli

FECHA DE ENTREGA: 27/10/23

FUNCIONAMIENTO DEL CÓDIGO

Importar módulos y definir constantes:

```
from trainer import train_model
from comparison import comparar_archivo, mostrar_data

dataset_path = "../minidataset"
nuevo_archivo = "../minidataset/TP2 V1.docx"
```

El módulo trainer contiene las clases y funciones necesarias para crear el modelo.
El módulo comparison realiza el análisis del plagio.
dataset_path define la ruta donde se encuentra el dataset.
nuevo_archivo define la ruta donde se encuentra el archivo a analizar.

Entrenar el modelo:

```
: model = train_model(dataset_path)
```

Devuelve un objeto de clase Model, el cual contiene un listado de profesores (hardcodeado, como se explicará más adelante) y un listado de diccionarios llamados tp, los cuales contienen:

- "filename": el nombre del archivo
- "text": el texto ya procesado del archivo

El texto procesado del archivo se obtiene utilizando la función file_reader del módulo reader, para después procesarlo (es decir, limpiarlo de caracteres extraños y pasarlo a minúsculas).

Realizar la comparación con el nuevo archivo:

```
analizado = comparar_archivo(nuevo_archivo, model)
```

Devuelve un objeto de clase Analysis, con el nombre del archivo, el título del trabajo, los profesores, los autores, el porcentaje de plagio en base a las coincidencias con otros textos, el listado de frases con una coincidencia de más del 50% (posiblemente plagiadas).
´

Mostrar la información pedida en la consigna:

```
mostrar_data(analizado)
```

Esta información es:

- Nombre del archivo
- Título/tópico del trabajo
- Profesores
- Autores
- Porcentaje de plagio
- Frases coincidentes

DECISIONES DE DISEÑO MÁS DESTACADAS

Hardcodeo de los profesores:
Identificar los nombres en un texto es muy costoso computacionalmente, ya que tiene que verificar que cada palabra no pertenezca al diccionario. Por lo tanto, la idea de identificar a los profesores al ver qué nombres se repiten frecuentemente recorriendo todos los trabajos prácticos queda descartada. Por eso se decidió guardar los posibles nombres de los profesores (incluyendo variaciones como que lleve o no segundo nombre, o sin acentos) en una lista.

Búsqueda de nombres:
La búsqueda de nombres (tanto de profesores como de alumnos) se realiza buscando entre las primeras palabras del texto, las que no se encuentren en el diccionario español (proporcionado por NLTK).

Obtención del título:
Se realiza buscando la primera oración del texto que no sea nombre, ni parte de una lista de palabras específica (relacionadas con elementos que podría haber en la portada, como "legajo", "alumno", "universidad tecnológica nacional", etc.).

Medición de la distancia entre textos y el plagio entre oraciones:
Para medir la distancia entre los textos, se utilizan las diversas herramientas que provee NLTK. Primero se separa cada texto en oraciones, y luego por cada una de estas se analiza la distancia entre ellas, utilizando la distancia jaccardiana. Esta distancia analiza el número de coincidencias exactas en un conjunto de palabras (la separación de cada oración en palabras). Se eligió esta distancia porque requiere que las palabras en el conjunto sean iguales, lo cual nos sirve porque si analizamos letra por letra podría tomar poca distancia con palabras similares aunque semánticamente significan distintas cosas. Si la distancia entre oraciones es de menos de 0.5 (es decir que coinciden en más de un 50%), se las añade a una lista de oraciones plagiadas, identificando el texto al que estas pertenecen. Con esta distancia, que se va sumando por cada oración, se obtiene la distancia total entre cada texto y se suma para ver al final el porcentaje de plagio del texto proporcionado.

Lectura de archivos:
Para el propósito de leer los archivos, se utilizó una biblioteca llamada textract, la cual sirve para los formatos de archivo proporcionados en el dataset. De este modo no nos preocupamos por el formato del mismo y lo leemos.

ASPECTOS A MEJORAR

Profesores:
Para poder leer los profesores dentro del dataset completo, se necesitaría: o una mayor capacidad computacional, o estandarizar el formato de las portadas de los trabajos del dataset.

Nombres:
Por la complejidad computacional que supondría no se hizo, pero para identificar mejor los nombres se podría aumentar el número de palabras en la lista de palabras del diccionario, incluyendo verbos conjugados y palabras del inglés. También podría estandarizarse el formato para listar los autores del trabajo, además de los profesores.

Títulos:
Al igual que en los puntos anteriores, para mejorar la detección de títulos sería necesaria una estandarización que se podría realizar para que estén de una manera determinada en la portada o en el nombre del archivo.

Lectura de archivos:
Este aspecto podría mejorar si solamente se aceptaran documentos en formato pdf, ya que hay ciertos formatos (como el .doc) que ocasionan errores al convertirlos en texto.

BIBLIOGRAFÍA

Biblioteca Natural Language Toolkit. (Año). NLTK 3.8.1 documentation. Recuperado de https://www.nltk.org/