Documentación de wrapper de acceso a corpus digitales

1. Introducción

Se presenta la documentación de un wrapper de acceso a la información contenida en corpus digitales de lenguas amazónicas de escasos recursos. Se utilizan seis clases, cada una con sus respectivos atributos y métodos de procesamiento y acceso.

2. Corpus

Clase principal para gestionar un corpus de datos textuales. Permite agregar, leer, y limpiar entradas del corpus.

2.1. Atributos

- root_directory (str): Directorio raíz donde se almacenan los archivos del corpus.
- encoding (str): Codificación de los archivos del corpus para su lectura.
- lengua_principal (str): Lengua principal del corpus.
- n_entries (int): Número de entradas en el corpus.
- entries (list de CorpusEntry): Lista de entradas que conforman el corpus.
- text_column (str): Columna del archivo que contiene el texto principal en un archivo JSON.
- file_column (str): Columna del archivo que contiene el nombre o referencia del archivo en un archivo JSON.
- mb_column (str): Columna del archivo que contiene las descomposiciones morfológicas en un archivo JSON. Los morfemas deben estar separados por espacios en blanco. El primer morfema de una palabra debe estar libre, los posteriores deben tener un carácter '=' o '-' para poder ser procesados.
- pos_column (str): Columna del archivo que contiene las etiquetas POS en un archivo JSON. Las etiquetas deben estar separadas por espacios en

blanco. La primera etiqueta de una palabra debe estar libre, las posteriores deben tener un carácter ‘=’ o ‘-‘ para poder ser procesadas.

- id_column (str): Columna del archivo que contiene el identificador de la entrada. tokenizer: Tokenizador utilizado para procesar las entradas en un archivo JSON.

2.2. Métodos

- __init__(self, root_directory, link=None, encoding="utf-8", ...): Inicializa un objeto Corpus.

  Parámetros:

  - root_directory (str): Directorio raíz del corpus.
  - encoding (str, opcional): Codificación de los archivos (por defecto es "utf-8").
  - lengua_principal (str, opcional): Lengua principal del corpus.
  - Otros parámetros se utilizan para definir las columnas que corresponden al texto, archivo, etiquetas POS, descomposiciones morfológicas y el identificador de las entradas, de acuerdo con los atributos de la clase.

- __str__(self): Retorna una representación legible del corpus mostrando todas las entradas.

- add_entry(self, entry): Agrega una nueva entrada al corpus.

  Parámetros:

  - entry (CorpusEntry): La entrada a agregar al corpus.

- read(self, file): Lee un archivo en formato JSON línea por línea, extrae las columnas relevantes para crear entradas de tipo CorpusEntry.

  Parámetros:

- file (str): Ruta del archivo a leer.

- clean(self, process_words=True, remove_duplicates=True, min_length=1): Limpia el corpus: elimina duplicados, procesa palabras y elima entradas cortas.

  Parámetros:

  - process_words (bool, opcional): Si es True, procesa las palabras de cada entrada (por defecto es True).
  - remove_duplicates (bool, opcional): Si es True, elimina entradas duplicadas basadas en su texto (por defecto es True).
  - min_length (int, opcional): Longitud mínima en número de palabras para considerar una entrada (por defecto es 1).

3. Corpus Entry

Esta clase representa una entrada en el corpus, asociando un texto principal con una lista de palabras y un identificador único.

3.1. Atributos

- file (str): Nombre o referencia del archivo que contiene la entrada del corpus.
- text (str): Texto principal de la entrada.
- words (list de WordEntry): Lista de objetos WordEntry que representan las palabras dentro del texto.
- id (str): Identificador único de la entrada.

3.2. Métodos

- __init__(self, file, text, words=None, entry_id=None): Constructor para inicializar un objeto CorpusEntry.

  Parámetros:

- file (str): Archivo de origen de la entrada.
- text (str): Texto que representa la entrada del corpus.
- words (list, opcional): Lista de palabras (objetos WordEntry). Si no se proporciona, se inicializa como una lista vacía.
- entry_id (str, opcional): Identificador de la entrada. Si no se proporciona, se genera uno automáticamente con uuid.

- __str__(self): Método que retorna una representación en cadena del objeto CorpusEntry.
- process_words(self): Procesa la lista de palabras de la entrada, llama al método process de WordEntry cuando hay solo una palabra en la lista y la reemplaza por el resultado del procesamiento.

4. Word entry

Representa una palabra dentro de una entrada del corpus, incluye información morfológica y gramatical opcional.

4.1. Atributos

- word (str): La palabra representada.
- mb (list o str): Descomposición morfológica de la palabra (morpheme breaks). Puede ser una lista o una cadena.
- pos (list o str): Etiquetas de partes del discurso (POS). Puede ser una lista o una cadena.

4.2. Métodos

- __init__(self, word, mb=None, pos=None): Inicializa un objeto WordEntry.

  Parámetros:

  - word (str): La palabra que se representa.

    - mb (list o str, opcional): Descomposición morfológica de la palabra. Si no se proporciona, se inicializa como una lista vacía.
    - pos (list o str, opcional): Etiquetas POS de la palabra. Si no se proporciona, se inicializa como una lista vacía.
- __str__(self): Retorna una representación en cadena de la palabra, junto con su descomposición morfológica (mb) y etiquetas POS (pos).
- process(self): Tokeniza la palabra utilizando espacios y signos de puntuación, y luego mapea la descomposición morfológica y las etiquetas POS a cada token. Mapea las etiquetas pos y morfemas a una misma palabra solo si las posteriores empiezan con '-' o '='.
    - Retorna una lista de nuevos objetos WordEntry, donde cada token es una palabra separada con sus respectivos morfemas y etiquetas POS.

5. Multilingual corpus

Extiende la clase Corpus para manejar datos multilingües con traducciones libres y glosas.

5.1. Atributos

- Hereda todos los atributos de Corpus.
- languages (list): Lista de lenguas adicionales en el corpus.
- gloss_columns (dict): Diccionario que mapea cada lengua a su columna de glosas.
- ft_columns (dict): Diccionario que mapea cada lengua a su columna de traducciones libres (free translations).

5.2. Métodos

- __init__(self, root_directory, link=None, ...): Inicializa un objeto MultilingualCorpus.

Parámetros:

- Hereda los parámetros de Corpus, con la adición de languages para definir los idiomas adicionales, y gloss_columns y ft_columns para definir las columnas asociadas a glosas y traducciones libres.

- add_entry(self, entry): Agrega una nueva entrada multilingüe al corpus.

  Parámetros:

  - entry (MultilingualCorpusEntry): La entrada multilingüe que se va a agregar. Debe ser de tipo MultilingualCorpusEntry.

- read(self, file): Lee un archivo JSON y extrae las columnas multilingües para crear entradas de tipo MultilingualCorpusEntry.

  Parámetros:

  - file (str): Ruta del archivo a leer.

- clean(self, process_words=True, remove_duplicates=True, min_length=1): Limpia el corpus multilingüe, con el procesamiento adicional de traducciones libres en las diferentes lenguas.

  Parámetros:

  - process_words (bool, opcional): Si es True, procesa las palabras de cada entrada multilingüe (por defecto es True).
  - remove_duplicates (bool, opcional): Si es True, elimina entradas duplicadas basadas en su texto (por defecto es True).
  - min_length (int, opcional): Longitud mínima en número de palabras para considerar una entrada (por defecto es 1).

6. Multilingual corpus entry

Extiende CorpusEntry para manejar las traducciones libres en múltiples lenguas.

6.1. Atributos

- Hereda los atributos de CorpusEntry
- ft (dict): Diccionario con traducciones libres, donde cada clave es un idioma y cada valor es la traducción en esa lengua.

6.2. Métodos

- __init__(self, file, text, words=None, entry_id=None, ft=None): Inicializa una entrada multilingüe.

  Parámetros:

  - Hereda los parámetros de CorpusEntry, con la adición de ft, que es un diccionario de traducciones libres. Si no se indica, se inicializa como un diccionario vacío.

- __str__(self): Retorna una representación legible de la entrada multilingüe, mostrando el texto principal y las traducciones libres.
- process_words(self): Procesa las palabras de la entrada multilingüe, llamando al método process en cada objeto de tipo MultilingualWordEntry.

7. Multilingual Word entry

Extiende WordEntry para manejar las glosas en múltiples lenguas.

7.1. Atributos

- Hereda los atributos de WordEntry.
- gloss (dict): Diccionario con glosas en múltiples lenguas.

7.2. Métodos

- __init__(self, word, mb=None, pos=None, gloss=None): Inicializa una palabra con glosas en varias lenguas.

  Parámetros:

    - Hereda los parámetros de WordEntry, con la adición de gloss, que es un diccionario de glosas. Si no se indica, se inicializa como un diccionario vacío.
- __str__(self): Retorna una representación legible de la palabra con sus glosas.
- process(self): Procesa la entrada multilingüe, tokeniza por palabras y las separa en subpalabras, y luego asocia las glosas con cada token.