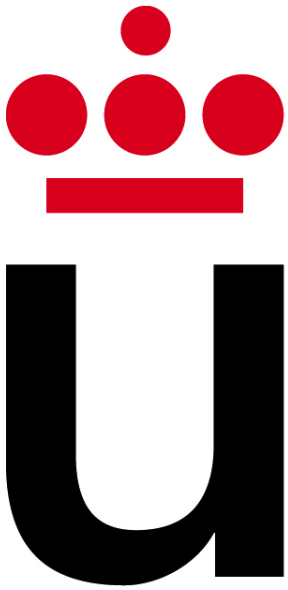
Paradigmas de Programación

Práctica 2

Ruby

GII+GIS: Luis Guillermo León Gámez



Móstoles, 11 de diciembre de 2015

Índice

1. Introducción **3**

2. Clases **3**
2.1. Main.rb . . . 3
2.2. Document.rb . . . 4
2.3. Scientific.rb . . . 4
2.4. Description.rb . . . 4
2.5. Section.rb . . . 4
2.6. Acronym.rb . . . 4
2.7. Cluster.rb . . . 5

3. Módulos **5**
3.1. Utils.rb . . . 5
3.2. Reader.rb . . . 5
3.2.1. read . . . 5
3.2.2. read_document . . . 5
3.2.3. read_section . . . 5
3.2.4. search . . . 6
3.2.5. clusterize . . . 6

4. Extensiones **6**

5. Comentarios **7**

1. Introducción

En esta práctica se pide hacer un programa en Ruby que sea capaz de procesar una colección de documentos a con el fin de conocer diferente información relacionada con ellos.
Los diferentes artículos que se proporcionan se corresponden con artículos científicos y descripciones de enfermedades de Wikipedia escritos en inglés y relacionados con la investigación sobre Enfermedades Raras. En todos los artículos aparece el nombre de una enfermedad rara acompañada de su acrónimo al menos una vez.

2. Clases

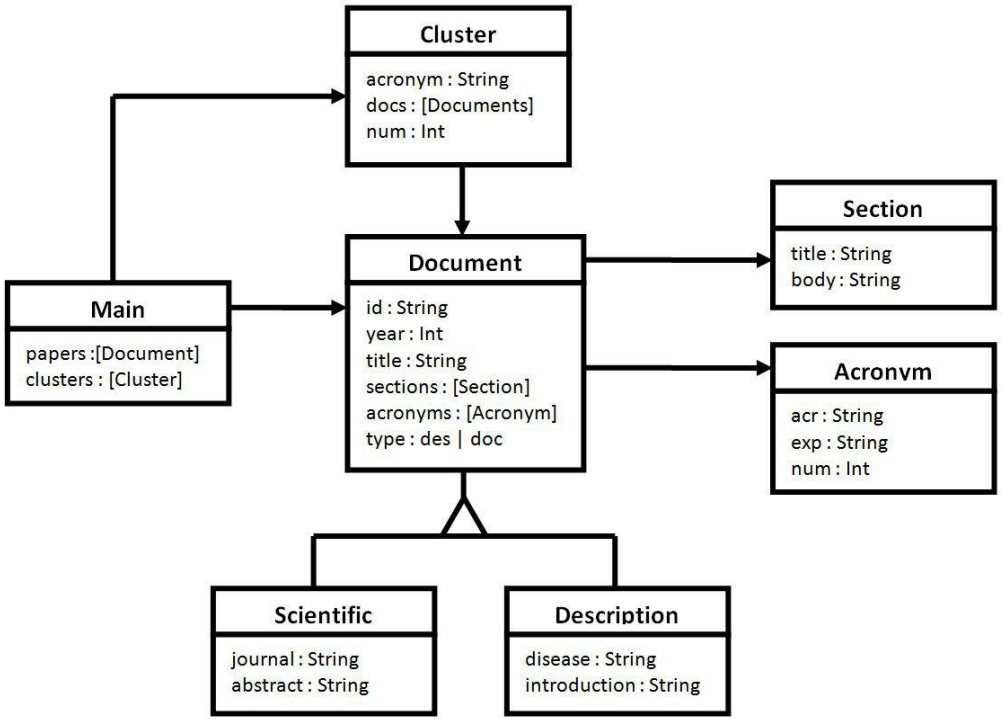


Figura 1: *Diagrama de clases*

2.1. Main.rb

Esta clase se utiliza para mostrar el menú principal y ejecutar las distintas opciones del mismo. Entre ellas cada opción enumerada se corresponde a su respectivo ejercicio, y las opciones entre paréntesis son funcionalidades adicionales.

2.2. Document.rb

Esta clase se corresponde con cualquier documento que se vaya a leer. Atributos:

- id :: String
- year :: Int
- title :: String
- sections :: [Section]
- acronyms :: [Acronym]
- type* :: String

*type:: des - Descripción de Wikipedia, doc - Documento científico.

2.3. Scientific.rb

Esta clase se corresponde con los documentos científicos. Hereda de *Document.rb*, e incorpora los siguientes atributos:

2.4. Description.rb

Esta clase se corresponde con las descripciones de enfermedades obtenidas de Wikipedia. Hereda de *Document.rb*, e incorpora los siguientes atributos

- disease :: Acronym
- introduction :: String

2.5. Section.rb

Esta clase se corresponde con las posibles secciones que puede tener un documento. Solo posee dos atributos:

- title :: String
- body :: String

2.6. Acronym.rb

Esta clase se corresponde con un acrónimo de un texto, y se posee los siguientes atributos:

- acr :: String
- exp :: String
- num :: Int

El atributo "acr" es el acrónimo en sí, "exp" es la forma expandida y "num" es el número de veces que aparece el acrónimo en el texto.

2.7. Cluster.rb

Esta clase se corresponde con una agrupación de documentos relacionados. Posee los siguientes atributos:

- acronym :: String
- docs :: [Document]
- num :: Int

El "atributo" acronym es el acrónimo encontrado que relaciona los dostintos documentos de la agrupación, "docs" es la lista de documentos de la agrupación y "num" es el número de documentos de la agrupación.

3. Módulos

3.1. Utils.rb

Este módulo contiene métodos y constantes genéricos necesarios en las distintas clases. El principal objetivo de este módulo es simplificar el código

3.2. Reader.rb

Este módulo contiene los métodos encargados de leer los documentos desde archivo e instanciar una clase Scientific o Description dependiendo del documento leido. También se encarga de sacar los acrónimos del texto, de agrupar los documentos.

3.2.1. read

Este método recibe el path del directorio donde se encuentran los archivos, y devuelve una lista de documentos.

3.2.2. read_document

Este método recibe el path de un documento y lo lee. Se distingue si un documento es un artículo científico o una descripción comprobando si la tercera línea es una cifra (en cuyo caso es un documento) o no (en cuyo caso es una descripción).

3.2.3. read_section

Tras haber leído la cabecera del documento (ya fuera artículo o descripción), este método divide el resto del texto en secciones (diferenciadas por la línea "–") y devuelve una lista con dichas secciones.

3.2.4. search

Este método se encarga de realizar la búsqueda de acrónimos. Para ello se han usado expresiones regulares. El proceso ha sido el siguiente: Primero se buscan todos los acrónimos en el texto que cumplan la siguiente expresión regular:

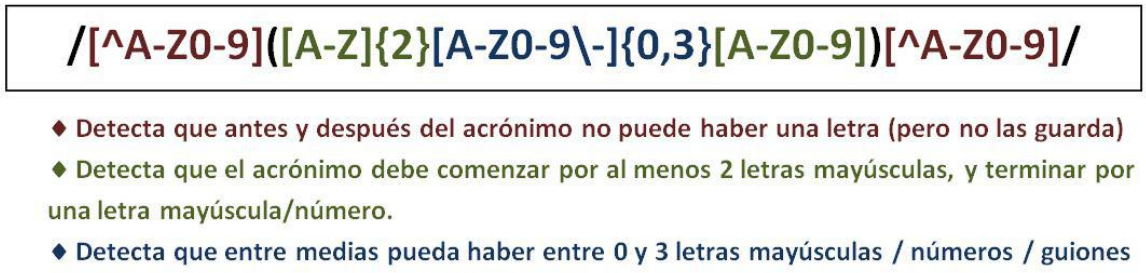


Figura 2: *Expresión regular - Lectura acrónimos*

Después, para buscar las formas expandidas de los acrónimos, creamos una expresión regular única por cada acrónimo siguiendo el siguiente algoritmo:

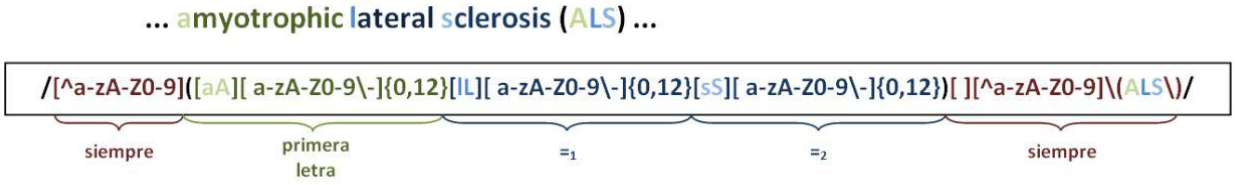


Figura 3: *Expresión regular - Ejemplo algoritmo forma expandida*

Con la expresión regular de la forma expandida, se busca la primera que concuerde en el texto. Tras ello contamos los duplicados y los eliminamos, creando con estos datos el acrónimo.

3.2.5. clusterize

Este método agrupa los documentos según su semejanza. La manera de escoger si dos documentos son parecidos es comparando los acrónimos más usados de cada documento (siguiendo la misma pauta de la práctica de haskell anterior).

4. Extensiones

Como se ha mencionado, existen varias funcionalidades más en el menú principal:

- **(complete)**: Como no siempre aparecen las formas expandidas de los acrónimos, esta función busca los acrónimos de un documento en el resto de documentos para encontrar su forma expandida (que puede haber sido encontrada en otro documento). Con ello se aumenta el número de acrónimos con forma expandida por documento.
- **(docs)**: muestra todos los documentos científicos.
- **(docs)**: muestra todas las descripciones de enfemedades.
- **(all)**: Muestra uno a uno la información de los artículos científicos (type, journal, title, year, abstract, section_number, sections y acronyms) y de las descripciónes de enfermedades (type, disease, year, introduction, section_number, sections y acronyms). Se puede avanzar escribiendo "n" (o pulsando enter), retroceder escribiendo "p", ir a un determinado documento escribiendo el número, y salir escribiendo "stop".
- **(top)**: Muestra la lista de documentos con sus acrónimos más usados.

5. Comentarios

Para ejecutar el código basta con abrir la terminal en la carpeta donde se encuentran el archivos *.rb* y ejecutar el siguiente comando:

```
ruby Main.rb
```