GPLSI Wikipedia Characterisation V1.0: Descubrimiento y Vinculación de Entidades a Wikipedia

GPLSI Wikipedia Characterisation V1.0: Entity Discovery and Linking to Wikipedia

Resumen de la aplicación: GPLSI Wikipedia Characterisation (Descubrimiento y vinculación de entidades a Wikipedia) constituye una interfaz de programación de aplicaciones (API) que incluye librerías de programación útiles para sistemas de terceros. Esta API ofrece la funcionalidad de analizar contenidos textuales para descubrir menciones de entidades y enlazarlas a Wikipedia mediante el uso de DBpedia, su versión estructurada. Como resultado se obtiene una lista de sugerencias de URIs de DBpedia (cada URI se corresponde con una página de Wikipedia) por cada entidad, ordenadas por el grado de confianza (en el intervalo [0,1]). Este grado de confianza se obtiene considerando dos características claves. La primera se corresponde con el número de enlaces entrantes para cada entidad de Wikipedia (más enlaces implica mayor relevancia). La segunda característica es la similitud entre el contexto (lista de palabras adyacentes a la palabra objetivo) de la entidad objetivo y la descripción de cada entidad de Wikipedia. Para este propósito se ha utilizado un algoritmo de desambiguación basado en el paradigma LESK, combinado con estadísticas sobre los enlaces entrantes a las páginas de Wikipedia. Los resultados que alcanza esta tecnología rondan el 70% de F1. La descripción completa del sistema está detallada en [TGA15] .

Summary: GPLSI Wikipedia Characterisation (Entity Discovery and Linking to Wikipedia) is an application programming interface (API) which programming libraries for third-parties. This service allows analysing textual content to discover Wikipedia entities related to that content by means of DBpedia, its structured version. As a result, a list of URIs from DBpedia (each one corresponding to a Wikipedia page) is obtained for each entity, ranked by a confidence score (in the interval [0,1]). This score is obtained considering two key features. The first one is the number of incoming links to the Wikipedia article (more links implies more relevance). The second one is the similarity of the context (list of words adjacent to the target word) of the entity found in text and the description of that entity in Wikipedia. For this purpose, the Lesk disambiguation algorithm has been followed, combined with statistics based on Wikipedia inlinks. The results achieved reflect around 70% of F1. The results achieved reflect around 70% of F1. A detailed description of the whole system can be found in [TGA15].

Especificaciones técnicas/ Technical Specifications

Lenguaje de programación/ Development Language: Java

Entorno Operativo/ Operating Environment: Linux/ Windows

Versión/Version: 1.0

Estructura de ficheros/ Files' structure:

Directorio/Folder -> data-characterisation-core:

- nbactions-characteriseDBPedia.xml
- nbactions.xml
- nb-configuration.xml
- pom.xml

Directorio/Folder -> data-characterisation-core/src/main/resources/models:

- en-chunker.bin
- en-ner-date.bin
- en-ner-money.bin
- en-ner-percentage.bin
- en-ner-time.bin
- en-pos-maxent.bin
- en-sent.bin
- testmallet.txt
- en-ner-asset.bin
- en-ner-location.bin
- en-ner-organization.bin
- en-ner-person.bin
- en-parser-chunking.bin
- en-pos-perceptron.bin
- en-token.bin

Directorio/Folder -> data-characterisation-core/src/main/java/es/ua/datacharacterisation:

- Main.java
- **Subdirectorio/Subfolder -> analyzer**
  - Definition.java
  - MorphoSyntacticAnalysis.java
  - POSTagger.java
  - TextEntity.java
- **Subdirectorio/Subfolder -> characterisation**
  - Coordinate.java
  - DbPediaRelatedResource.java
  - DbPediaResource.java
  - DBPediaResponse.java
  - **Subdirectorio/Subfolder -> controller**
    - CharacterisationController.java
    - DbPediaCharacteriser.java
- **Subdirectiorio/Subfolder -> characterise**
  - Characteriser.java
- **Subdirectiorio/Subfolder -> contextsimilarity**
  - ContextSimilarityFactory.java
  - ContextSimilarityStrategy.java
  - LevenshteinStemmerStrategy.java
  - LevenshteinStrategy.java
  - WordPercentageStemmerStrategy.java
  - WordPercentageStrategy.java
- **Subdirectorio/ Subfolder -> exception:**
  - DataCharacterisationException.java
  - InvalidParametersException.java
- **Subdirectorio/Subfolder -> response:**
  - ProfilerCharacteriseDBPediaResponse.java
  - RelatedDBPediaResponse.java
- **Subdirectorio/Subfolder -> util:**
  - DBpediaLookupClient.java
  - HttpClient.java
  - LevenshteinCalculator.java
  - LookUpEntity.java
  - Pair.java
  - PorterStemmer.java

Directorio/Folder -> data-characterisation-web:

- nb-configuration.xml
- pom.xml

Directorio/ Folder ->

data-characterisation-web/src/main/java/es/ua/datacharacterisation/application:

- ApplicationConfig.java

Directorio/ Folder -> data-characterisation-web/src/main/java/es/ua/datacharacterisation/services:

- DBPediaCharacterisation.java
- **Subdirectorio/Subfolder -> characterise**
  - CharacteriseParameters.java
- **Subdirectorio/Subfolder -> examples**
  - CharacterisationExamples.java
- **Subdirectorio/Subfolder -> filters:**
  - EntityFinderFilter.java
- **Subdirectorio/Subfolder -> internal/annotation**
  - ApplicationInfo.java
- **Subdirectorio/Subfolder -> internal/utils**
  - AnnotatedTextProcessor.java
  - ReflectionUtils.java
- **Subdirectorio/Subfolder -> listener**
  - ContextListener.java
- **Subdirectorio/Subfolder -> parameters**
  - IParameters.java
- **Subdirectorio/Subfolder -> security**
  - SecurityUtils.java
  - UntrustedAccessExceptionMapper.java
  - UntrustedAccessException.java

Directorio/ Folder -> data-characterisation-web/src/main/resources/models:

- en-chunker.bin
- en-ner-date.bin
- en-ner-money.bin
- en-ner-percentage.bin
- en-ner-time.bin
- en-pos-maxent.bin
- en-sent.bin
- testmallet.txt
- en-ner-asset.bin
- en-ner-location.bin
- en-ner-organization.bin
- en-ner-person.bin
- en-parser-chunking.bin
- en-pos-perceptron.bin
- en-token.bin

Directorio / Folder -> data-characterisation-web/src/main/webapp:

- entityfinder.html
- index.html
- logs.html
- **Subdirectorio/Subfolder -> css**
  - bootstrap.min.css
  - bootstrap-theme.min.css
  - log-style.css
  - main.css
  - style.css
- **Subdirectorio/Subfolder -> doc**
  - **Subdirectorio/Subfolder -> css**
    - reset.css
    - screen.css
  - **Subdirectorio/Subfolder -> img**
    - bg.png
    - logo.png
    - pet_store_api.png
    - wordnik_api.png
    - explorer_icons.png
    - logo_small.png
    - throbber.gif
  - index.html.bak
  - o2c.html
  - swagger-ui.min.js
  - favicon.ico
  - index.html
  - **Subdirectorio/Subfolder -> lib**
    - backbone-min.js
    - jquery.slideto.min.js
    - swagger.js
    - handlebars-1.0.0.js
    - jquery.wiggle.min.js
    - swagger-oauth.js
    - handlebars-2.0.0.js
    - marked.js
    - underscore-min.js
    - highlight.7.3.pack.js
    - **Subdirectorio/Subfolder-> shred**
      - content.js
    - underscore-min.map
    - jquery-1.8.0.min.js
    - shred.bundle.js
    - jquery.ba-bbq.min.js
    - swagger-client.js
    - swagger-ui.js
- **Subdirectorio/Subfolder -> fonts**
  - glyphicons-halflings-regular.eot
  - glyphicons-halflings-regular.ttf
  - glyphicons-halflings-regular.svg
  - glyphicons-halflings-regular.woff
- **Subdirectorio/Subfolder -> img**
  - bg-header.png

    - bg.png
    - gplsi.png
- **Subdirectorio/Subfolder ->js**
    - ajax.js
    - bootstrap.min.js
    - jquery-2.1.1.min.js
    - logs.js
    - app.js
    - **Subdirectorio/Subfolder -> controllers**
        - main.js
    - jquery-ui.js
- **Subdirectorio/Subfolder -> META-INF**
    - context.xml
- **Subdirectorio/Subfolder -> views**
    - main.html
    - **Subdirectorio/Subfolder -> fragments**
        - footer.html
        - header.html
- **Subdirectorio/Subfolder -> WEB-INF**
    - host.properties
    - log4j.properties
    - security.properties
    - test.txt
    - web.xm

Requerimientos/Requirements: Java 1.7 o superior instalado, 3 GB de RAM o superior, 1 GB de disco duro libre en sistema, Apache Tomcat 8.0 o superior, Maven y conexión a Internet para acceder a DBpedia. Además, son necesarios los modelos de Apache OpenNLP. Dichos modelos ya se incluyen en la distribución de la aplicación por comodidad del usuario, pero no necesariamente se tiene que distribuir dentro de ésta, ya que son recursos externos[1].

Instalación/Installing: Descargar el código fuente del repositorio de control de versiones. Se deben generar los binarios tanto de la aplicación de consola como la web utilizando Maven. Antes de generar los binarios, es importante asegurarse de que los modelos de Apache OpenNLP se encuentran en los directorios [core|web]/src/main/resources/models.

En este momento la aplicación de consola está lista para funcionar.

La instalación de la aplicación web es opcional. Además de lo anterior, es necesario desplegar el war generado en el servidor de aplicaciones Tomcat.

En principio se puede usar en cualquier sistema operativo, pero sólo ha sido probado en Ubuntu Linux.

Ejecución/Run: La aplicación web dispone de una página de documentación del servicio web REST. El servicio se puede ejecutar directamente desde esta documentación en un navegador o mediante el uso clientes para el servicio web REST.

La aplicación de consola dispone de un comando para caracterizar un texto o un archivo (nunca ambos) sobre DBpedia, así como un comando de ayuda. A continuación mostramos algunos ejemplos:

- $**[nombre del jar] help characterisedbpedia**
    - Muestra la ayuda del comando, con los parámetros y su descripción.
- **$[nombre del jar] characterisedbpedia -f/--file [nombre del archivo txt]**

[1]Disponibles en http://opennlp.sourceforge.net/models-1.5/

- Permite proporcionar un archivo txt con el texto que se desea utilizar.

Dependencias/Dependencies:

El listado completo de dependencias, tanto para la aplicación de consola como para el servicio web, se encuentra en los archivo pom.xml. Aquí se incluye un listado de las dependencias que se descargarán de varios repositorios Maven, en el cual las librerías propias del sistema están marcadas con un *:

Core

- airline 0.7
- commons-httpclient 3.1
- commons-io 2.4
- commons-lang 3.1
- commons-codec 1.2
- commons-logging 1.0.4
- gson 2.3
- guava 19.0
- javax.ws.rs-api 2.0.1
- jersey-guava 2.15
- junit 4.12
- lucene-analyzers-common 4.9.0
- opennlp tools 1.5.3
- swagger-annotations 1.3.11
- xml-utils 1.0.0
- annotations 2.0.3
- hamcrest-core 1.3
- javax.inject 1
- jwnl 1.3.3
- lucene-core 4.9.0
- opennlp maxent 3.0.3

Web

- data-characterisation-core 1.0.0*
- gson 2.3
- guava 15.0
- jackson-annotations 2.5.0
- jackson-core 2.5.0
- jackson-databind 2.5.0
- jackson-jaxrs-base 2.5.0
- jackson-jaxrs-json-provider 2.5.0
- jackson-jaxrs-xml-provider 2.5.0
- jackson-module-jaxb-annotations 2.5.0
- javaee-web-api 7.0
- javassist 3.18.1-GA
- javax.annotation-api 1.2
- javax.servlet-api 3.0.1
- javax.ws.rs-api 2.0.1
- jaxb-api 2.2.7
- jersey-client 2.15
- jersey-common 2.15
- jersey-container-servlet 2.15
- jersey-container-servlet-core 2.15
- jersey-entity-filtering 2.15
- jersey-guava 2.15
- jersey-media-json-jackson 2.15
- jersey-media-multipart 2.15
- jersey-server 2.15
- log4j 1.2.17
- persistence-api 1.0
- swagger-annotations 1.3.11
- swagger-core_2.10-1.3.11
- swagger-core 1.5.3
- swagger-jaxrs_2.10-1.3.11
- swagger-jaxrs 1.5.3
- swagger-jersey2-jaxrs_2.10-1.3.11
- swagger-jersey2-jaxrs 1.5.3
- swagger-utils_2.10-1.3.11
- swagger-models 1.5.3
- xom 1.2.10
- airline 0.7
- annotations 2.0.3
- aopalliance-repackaged 2.4.0-b06
- commons-httpclient 3.1
- commons-io 2.4
- commons-lang 3.1
- commons-lang 2.4
- commons-codec 1.2
- commons-logging 1.0.4
- dom4j 1.6.1
- hamcrest-core 1.3
- hk2-api 2.4.0-b06
- hk2-locator 2.4.0-b06
- hk2.hk2-utils 2.4.0-b06
- jackson-dataformat-xml 2.5.0
- jackson-module-jsonSchema 2.1.0
- jackson-module-scala 2.10-2.41
- javax.inject 2.4.0-b06
- javax.inject 1
- joda-convert 1.2
- joda-time 2.2
- json4s-ast_2.10 3.2.9
- json4s.json4s-core_2.10 3.2.9
- json4s.json4s-ext_2.10 3.2.9
- json4s.json4s-jackson_2.10 3.2.9
- json4s.json4s-native_2.10 3.2.9
- jsr305 2.0.1
- jsr311-api 1.1.1
- junit 4.12
- jwnl 1.3.3
- lucene-core 4.9.0
- lucene-queryparser 4.9.0
- lucene-analyzers-common 4.9.0
- lucene-sandbox 4.9.0
- mimepull 1.9.3
- opennlp-maxent 3.0.3
- opennlp-tools 1.5.3
- osgi-resource-locator 1.0.1
- paranamer 2.6
- reflections 0.9.9-RC1
- scala-compiler 2.10.0
- scala-library 2.10.0

- scala-reflect 2.10.4
- scalap 2.10.0
- slf4j-api 1.6.3
- stax-api 1.0.2
- stax2-api 3.1.1
- swagger-annotations 1.5.3
- woodstox-core-asl-4.4.0
- xalan 2.7.0
- xercesImpl 2.8.0
- xml-apis 1.3.03
- xml-utils 1.0.0

Referencias/References

[LES86] Lesk, M. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In SIGDOC '86: Proceedings of the 5th annual international conference on Systems documentation, pp 24-26, New York, NY, USA. ACM, 1986.

[TGA15] Tomás,D. Gutiérrez, Y. Agulló, F. Entity Linking in Media Content and User Comments: Connecting Data to Wikipedia and other Knowledge Bases. Proceedings of eChallenges 2015 e-2015. pp 1-10, 2015.